



# **UNIVERSITÀ DEGLI STUDI DI PADOVA**

**CORSO DI LAUREA MAGISTRALE IN  
INGEGNERIA ELETTRICA**

**TESI DI LAUREA MAGISTRALE**

**NUOVA IMPLEMENTAZIONE DI UN CIRCUITO A  
MICROCONTROLLORE PER LA GESTIONE DEI  
PARAMETRI ILLUMINOTECNICI E TERMICI DI UNA  
SORGENTE A LED CON CROMATICITÀ VARIABILE**

**RELATORE: Prof. Pietro Fiorentin**

**Dipartimento di Ingegneria Industriale**

**CORRELATORE: Ing. Alessandro Scroccaro**

**LAUREANDO: Simone Mele**

**ANNO ACCADEMICO 2015/2016**

Alla mia famiglia

## INDICE

<i>INTRODUZIONE</i> .....	10
<i>CAPITOLO 1 – ASPETTI TEORICI FONDAMENTALI</i> .....	11
1.1 – <i>L’occhio umano e la radiazione visibile</i> .....	11
1.2 – <i>Grandezze fotometriche fondamentali</i> .....	12
1.2.1 – <i>Flusso luminoso</i> .....	13
1.2.2 – <i>Luminanza</i> .....	14
1.2.3 – <i>Illuminamento</i> .....	15
1.3 – <i>Colorimetria</i> .....	17
1.3.1 – <i>Leggi di Grassman</i> .....	17
1.4 – <i>La tecnologia LED</i> .....	18
1.5 – <i>Modulazione della larghezza d’impulso (PWM)</i> .....	19
1.6 – <i>Sistema di controllo</i> .....	20
<i>CAPITOLO 2 – VISIONE DEL SISTEMA ELETTRICO PREESISTENTE – HARDWARE</i> ..	23
2.1 – <i>Hardware – Descrizione del sistema elettrico di partenza</i> .....	23
2.1.1 – <i>Sorgenti luminose impiegate</i> .....	23
2.1.2 – <i>Sfera Integratrice</i> .....	25
2.2 – <i>Circuito di alimentazione dei LED</i> .....	26
2.2.1 – <i>Misura di corrente inviata ai LED</i> .....	27
2.2.2 – <i>Alimentatori integrati dei LED</i> .....	28
2.2.3 – <i>Erogazione della tensione di controllo ai driver</i> .....	30
2.3 – <i>Sensori per la misura della frequenza</i> .....	31
2.4 – <i>Circuito per il controllo della temperatura</i> .....	33
2.4.1 – <i>Sensore per la misura di temperatura del sistema</i> .....	34
2.4.2 – <i>Riscaldatori del sistema</i> .....	36

2.4.3 – Sistema di raffreddamento .....	38
2.5 – Sistema a microcontrollore Arduino .....	41
<b>CAPITOLO 3 – IL NUOVO CIRCUITO ELETTRICO - HARDWARE – SOFTWARE .....</b>	<b>42</b>
3.1 – Considerazioni iniziali .....	42
3.2 - Hardware – Il nuovo circuito elettrico .....	42
3.3 – Settore 1 – Alimentazione e controllo dei LED .....	44
3.3.1 – Principio del partitore di tensione .....	47
3.3.2 – Misura di tensione ai capi degli alimentatori dc-dc .....	48
3.3.3 – Misura della caduta di tensione ai capi della resistenza di shunt .....	49
3.3.4 – Calcolo della tensione presente ai capi della serie dei LED .....	50
3.3.5 – Calcolo della tensione misurata sul CTRL degli alimentatori dei LED .....	51
3.3.6 – Misura di temperatura del sistema .....	52
3.4 – Settore 2 – Microcontrollore Arduino e collegamenti per le misure .....	53
3.5 – Settore 3 – Sistema di regolazione e controllo della temperatura .....	55
3.6 – Cavo di collegamento tra il nuovo circuito e il sistema contenente i LED .....	56
3.7 – Software – Modifiche apportate al firmware di Arduino .....	58
3.7.1 – Misura di tensione al pin CTRL dell'alimentatore dc-dc .....	59
3.7.1.2 – Modifiche apportate e problemi riscontrati .....	59
3.7.2 – Misura di tensione erogata dagli alimentatori dei LED .....	63
3.7.3 – Misura della caduta di tensione ai capi della resistenza di shunt .....	65
3.7.4 – Misura di tensione ai capi della serie di LED .....	66
3.7.5 – Introduzione del setpoint di temperatura .....	67
3.8 – Comandi per attivare le operazioni di misura e settaggio dei parametri .....	72
<b>CAPITOLO 4 - MISURA DELLA TEMPERATURA DI GIUNZIONE DEI LED .....</b>	<b>77</b>
4.1 – Struttura e principio di funzionamento di un LED .....	77

4.2 – Drogaggio di un semiconduttore e la giunzione P-N .....	78
4.3 – Polarizzazione della giunzione P-N ed emissione di fotoni .....	79
4.4 – Power LED .....	80
4.5 – La temperatura di giunzione $T_j$ .....	81
4.6 – Misura della temperatura di giunzione $T_j$ .....	83
4.7 – Stima della temperatura di giunzione con il nuovo sistema elettrico .....	85
4.8 – Codice MatLab per la misura di temperatura di giunzione $T_j$ .....	86
4.9 – Risultati ottenuti eseguendo le misure .....	90
5 – CONCLUSIONI E SVILUPPI FUTURI .....	95
5.1 – Conclusioni .....	95
5.2 – Sviluppi futuri .....	95
Listati MatLab Utilizzati .....	97
Firmware di Arduino .....	97
Il nuovo circuito elettrico implementato .....	98
Bibliografia .....	99

## INDICE DELLE FIGURE

<i>CAPITOLO 1 – ASPETTI TEORICI FONDAMENTALI</i> .....	11
<i>Fig. 1 – L’occhio</i> .....	11
<i>Fig.1.1 – Curve di risposta standard dell’occhio</i> .....	12
<i>Fig. 1.1.2 – Spettro di radiazioni elettromagnetiche</i> .....	12
<i>Fig. 1.2 – Sfera integratrice</i> .....	14
<i>Fig.1.2.2 – Superficie luminosa</i> .....	14
<i>Fig.1.2.3 – Luminanzometro</i> .....	15
<i>Fig.1.2.4 – Spettroradiometro</i> .....	15
<i>Fig.1.2.5 – Luxmetro</i> .....	17
<i>Fig.1.4 – Modello costruttivo di un LED</i> .....	18
<i>Fig.1.4.2 – Confronto tra spettri di emissione</i> .....	19
<i>Fig. 1.4.3 – LED di potenza (a) – Zona di svuotamento (b)</i> .....	19
<i>Fig.1.5 – Interruttore (a) e Caratteristica dell’onda di uscita modulata(b)</i> .....	20
<i>Fig.1.6 – Sistema controllato con retroazione</i> .....	20
<i>Fig.1.6.2 – Regolatore PID</i> .....	21
<i>CAPITOLO 2 – VISIONE DEL SISTEMA ELETTRICO PREESISTENTE – HARDWARE</i> ..	23
<i>Fig.2.1.1 – LED utilizzati</i> .....	24
<i>Fig.2.1.2 – Disposizione dei LED sul supporto circolare</i> .....	25
<i>Fig.2.1.3 – Sfera integratrice</i> .....	26
<i>Fig.2.2 – Circuito di alimentazione dei LED</i> .....	26
<i>Fig.2.2.2 – Alimentatore LuxDrive</i> .....	28
<i>Fig.2.2.3 – Caratteristica fornita dal costruttore</i> .....	28
<i>Fig.2.2.4 – Collegamenti dell’alimentatore</i> .....	29
<i>Fig.2.2.5 – Condensatori ceramici</i> .....	29

<i>Fig.2.2.6 – Caratteristica dell'alimentatore implementata</i> .....	30
<i>Fig.2.2.7 – Filtro RC</i> .....	30
<i>Fig.2.3 – Sensore TAOS</i> .....	31
<i>Fig.2.3.1 – Schema a blocchi del sensore</i> .....	31
<i>Fig.2.3.2 – Scheda di collegamento dei conduttori</i> .....	32
<i>Fig.2.3.4 – Sensore TAOS (a) – Componente di supporto (b)</i> .....	32
<i>Fig.2.3.4 – Supporto montato sulla sfera integratrice</i> .....	32
<i>Fig.2.4 – Schema circuitale per il controllo di temperatura</i> .....	33
<i>Fig.2.4.1 – Circuito elettrico del controllo di temperatura</i> .....	34
<i>Fig.2.4.1.1 – Trasduttore di temperatura</i> .....	34
<i>Fig.2.4.1.2 – Datasheet fornito dal costruttore</i> .....	35
<i>Fig.2.4.1.3 – Schema circuitali di collegamento del sensore</i> .....	35
<i>Fig.2.4.2 – Riscaldatore</i> .....	36
<i>Fig.2.4.2.1 – Caratteristica di temperatura della flangia</i> .....	37
<i>Fig.2.4.2.2 – Caratteristica di temperatura – potenza</i> .....	37
<i>Fig.2.4.2.3 – Datasheet fornito dal costruttore</i> .....	38
<i>Fig.2.4.2.4 – Riscaldatori</i> .....	38
<i>Fig.2.4.3 – Particolare costruttivo del controllo di temperatura</i> .....	39
<i>Fig.2.4.3.1 – Mosfett International Rectifier IRF530N</i> .....	39
<i>Fig.2.4.3.2 – Schema circuitale di collegamento del mosfett</i> .....	40
<i>Fig.2.4.3.3 – Collegamento del mosfett e della resistenza</i> .....	40
<i>Fig.2.5 – Microcontrollore Arduino Mega AT2560</i> .....	41
<i>CAPITOLO 3 – IL NUOVO CIRCUITO ELETTRICO - HARDWARE – SOFTWARE</i> .....	42
<i>Fig.3.1 – Circuito elettrico usato in precedenza</i> .....	42
<i>Fig.3.2 (a) – Nuovo schema circuitale</i> .....	43
<i>Fig.3.2 (b) – Nuovo circuito elettrico realizzato</i> .....	43

<i>Fig.3.2.1 – Settori compartimentali del circuito</i> .....	44
<i>Fig.3.3 – Settore 1</i> .....	45
<i>Fig.3.3.1 – Filtri e partitori per la misura</i> .....	47
<i>Fig.3.3.6 – Scheda contenente i filtri</i> .....	53
<i>Fig.3.4 – Settore 2</i> .....	53
<i>Fig.3.4.1 – Morsettiera di collegamento</i> .....	55
<i>Fig.3.5 – Settore 3</i> .....	55
<i>Fig.3.6 (a) Conduttori precedenti – (b) Nuovi collegamenti</i> .....	57
<i>Fig.3.6.1 – Spina di collegamento</i> .....	57
<i>Fig.3.7.1.2 – Caratteristica Corrente – Tensione</i> .....	60
<i>Fig.3.7.5 – Flow Chart del controllo di temperatura</i> .....	68
<i>Fig.3.8 – Menù dei comandi stampati a video dal terminale</i> .....	73
<b><i>CAPITOLO 4 - MISURA DELLA TEMPERATURA DI GIUNZIONE DEI LED</i></b> .....	77
<i>Fig.4.1 – Bande di Valenza, Conduzione e Band Gap</i> .....	77
<i>Fig.4.2 – Giunzione P-N (a) – Zona di Svuotamento (b)</i> .....	78
<i>Fig.4.3 – Polarizzazione ed emissione di fotoni</i> .....	79
<i>Fig.4.3.1 – Andamenti di distribuzione spettrale</i> .....	80
<i>Fig.4.4 – Power LED</i> .....	81
<i>Fig.4.5 – Resistenze termiche</i> .....	82
<i>Fig.4.5.1 – Caratteristica Tensione-Temperatura di giunzione</i> .....	82
<i>Fig.4.5.2 – Componenti di un LED</i> .....	83
<i>Fig.4.6 – Caratteristica Tensione – Temperatura – Corrente</i> .....	85
<i>Fig.4.7 – Caratteristica Tensione – Corrente</i> .....	86
<i>Fig. 4.9 (a) – Caratteristica Temperatura – N cicli</i> .....	92
<i>Fig.4.9 (b) – Caratteristica Corrente – Tensione ai capi di un LED con temperatura misurata costante</i> .....	93
<i>Fig.4.9 (c) – Caratteristica Tensione – Temperatura con corrente erogata costante</i> .....	94



## INDICE DELLE TABELLE

<i>CAPITOLO 2 – VISIONE DEL SISTEMA ELETTRICO PREESISTENTE – HARDWARE ..23</i>	
<i>Tab.2.1.1 – Caratteristiche luminose del LED .....</i>	<i>24</i>
<i>Tab.2.3.1 – Valori degli ingressi .....</i>	<i>32</i>
<i>CAPITOLO 3 – IL NUOVO CIRCUITO ELETTRICO - HARDWARE – SOFTWARE .....</i>	<i>42</i>
<i>Tab. 3.3 – Resistenze .....</i>	<i>45</i>
<i>Tab. 3.3.1- Condensatori .....</i>	<i>46</i>
<i>Tab.3.4 (a) - Pin di Arduino che individuano gli ingressi analogici .....</i>	<i>53</i>
<i>Tab.3.4 (b) - Pin di Arduino che individuano le uscite PWM .....</i>	<i>54</i>
<i>Tab.3.4 (c) - Pin di Arduino che individuano gli ingressi e le uscite digitali .....</i>	<i>54</i>
<i>Tab.3.5(a) - Tabella delle resistenze .....</i>	<i>56</i>
<i>Tab. 3.5 (b) - Tabella dei condensatori .....</i>	<i>56</i>
<i>Tab.3.6 – Collegamenti del cavo e delle spine .....</i>	<i>58</i>
<i>Tab.3.7.4 – Valori di tensione misurata con multimetro .....</i>	<i>67</i>
<i>Tab.3.8 – Caratteri e codifica ASCII .....</i>	<i>74</i>
<i>CAPITOLO 4 - MISURA DELLA TEMPERATURA DI GIUNZIONE DEI LED .....</i>	<i>77</i>
<i>Tab.4.3 – Materiali utilizzati per la giunzione P-N .....</i>	<i>80</i>
<i>Tab. 4.9 – Valori di tensione misurati con temperatura e corrente costante .....</i>	<i>91</i>

## INTRODUZIONE

Il lavoro di tesi esposto in questo elaborato è stato sviluppato a partire da un progetto iniziato quattro anni fa all'interno del laboratorio di illuminotecnica e fotometria dell'Università degli Studi di Padova, riguardante un sistema elettrico costituito da un hardware e un software, per la gestione e il controllo di una sorgente a cromaticità variabile, basato sul principio della somma additiva di colori tramite l'utilizzo della tecnologia *Light Emitting Diode* (LED).

Dalla preesistente struttura utilizzata per il funzionamento del sistema, il lavoro mira ad un ulteriore sviluppo degli aspetti che permettono di ottenere misure e impostazioni di parametri elettrici e illuminotecnici fondamentali.

Inizialmente il circuito si presenta attraverso tre diverse schede elettroniche stampate, collegate tra loro per mezzo di una sequenza di connettori rigidi o flessibili, di colore diverso. I connettori realizzano inoltre il collegamento fisico dei diversi dispositivi per la misura, con i circuiti e i morsetti di Arduino, il microcontrollore utilizzato per interfacciare il sistema con il terminale di controllo.

Le schede elettroniche sono collegate attraverso un fascio di conduttori a una piccola sfera integratrice realizzata in materiale plastico, costruita tramite stampa 3D e contenente i LED di diverso colore. I LED sono disposti secondo una struttura geometrica circolare su un supporto in alluminio con due resistori utilizzati come riscaldatori, una ventola utile al raffreddamento, un sensore per la misura della temperatura e un sensore composto di foto rilevatori in grado di misurare la frequenza di funzionamento da un segnale luminoso.

Il sistema circuitale impostato sull'utilizzo di tre differenti schede elettroniche, non permette il facile raggiungimento dei punti utilizzati come sorgenti per le misure, e influisce negativamente sulla possibilità di apportare modifiche in modo da migliorarne il funzionamento e lo sviluppo di nuovi sistemi di misura.

Durante la costruzione del nuovo circuito elettrico, sono state riscontrate differenze tra i valori dei precedenti elementi elettrici utilizzati e i nuovi, anche se questi ultimi presentano grandezze teoriche identiche. Questa diversità è dovuta al fatto che gli elementi presentano scostamenti dai valori dichiarati dovuti al metodo utilizzato per la loro costruzione dalle differenti case produttrici. Una volta tarato il nuovo sistema, è stato possibile procedere allo step successivo, introducendo i nuovi supporti hardware e software per il monitoraggio e il controllo delle nuove misure programmate per questo lavoro.

Questo lavoro di tesi si pone dunque come punto di partenza la necessità di migliorare la struttura circuitale preesistente e la conseguente necessità di migliorare il firmware utilizzato dal controllore Arduino, per permettere all'utente di selezionare la tipologia di misura o impostazione da far eseguire al circuito.

## CAPITOLO 1 – ASPETTI TEORICI FONDAMENTALI

In questo capitolo verranno presi in considerazione gli aspetti teorici fondamentali per capire l'interazione esistente tra una radiazione luminosa con l'organo in grado di elaborare l'informazione e il sistema matematico che esprimono le interazioni.

### 1.1- L'occhio umano e la radiazione visibile

Per avviare uno studio su qualsiasi campione di colore o radiazione luminosa è necessario capire come l'occhio, unico e solo organo in grado di adempiere a questa funzione, è composto e quali parti essenziali permettono di trasformare una radiazione luminosa in impulsi elettrici da inviare al cervello per essere interpretati. L'occhio Fig.1 umano è costituito principalmente dal *DIOTTRO* (cornea, umor vitreo e umor acqueo), e da una lente biconvessa, il *CRISTALLINO*, che permette la variazione della distanza focale attraverso l'utilizzo dei muscoli ciliari che agiscono modificando la curvatura della lente anteriore, migliorando la messa a fuoco dell'immagine campione. La cornea, la camera anteriore, il cristallino e la camera posteriore, formano nell'insieme una lente convergente che trasmette immagini sulla *RETINA*, rimpicciolite e capovolte. Attraverso un'apertura ricavata sulla superficie dell'*IRIDE* denominata

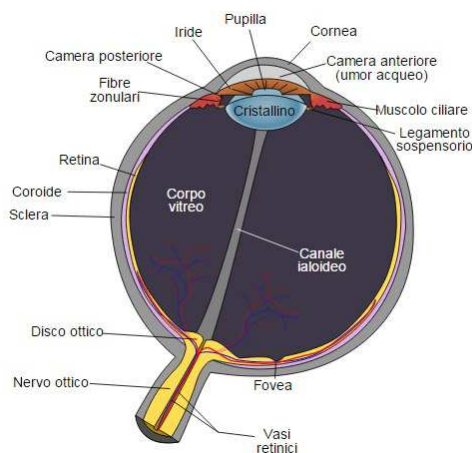


Fig. 1 – L'occhio [1]

*PUPILLA*, costituita da una membrana muscolare in grado di dilatarsi e contrarsi, è possibile modificare la quantità di luce entrante nell'occhio. La *RETINA*, costituente la parte più sensibile dell'occhio, è composta da una serie di recettori con determinate caratteristiche denominate *BASTONCELLI* e *CONI*, aventi lo scopo di trasformare in impulsi elettrici le reazioni fotochimiche sviluppate in funzione della radiazione luminosa ricevuta. Questi impulsi elettrici sono inviati ai *NEURONI RETINICI* i quali sviluppano una prima elaborazione del segnale che sarà inviata al nervo ottico e dunque al cervello per l'interpretazione finale. I bastoncelli e i coni costituiscono i fotorecettori più importanti ed efficaci in diverse condizioni di luminosità. I bastoncelli sono molto più utili per la visione *SCOTOPICA* (notturna) di quanto non lo siano i coni utilizzati dall'occhio fondamentalmente per la visione *FOTOPICA* (diurna) [2].

Dunque saranno i coni ad essere presi maggiormente in considerazione per lo studio dei fenomeni cromatici e saranno classificati secondo la loro sensibilità alle diverse lunghezze d'onda del segnale luminoso. I coni si possono distinguere in tre diverse categorie, a seconda della sensibilità spettrale:

- Lunghezza d'onda lunga – Luce Rossa (R)
- Lunghezza d'onda media – Luce Verde (G)
- Lunghezza d'onda corta – Luce Blu (B)

Da questa distinzione si nota come una determinata percezione cromatica, dipenda dall'interazione tra radiazioni luminose aventi differenti lunghezze d'onda pesate con la sensibilità percettiva dell'occhio umano, classificate dalla *Commission International*

d'Eclairage (CIE) nel 1931 attraverso le curve di risposta standard dell'occhio che definiscono l'osservatore standard CIE Fig. 1.1:

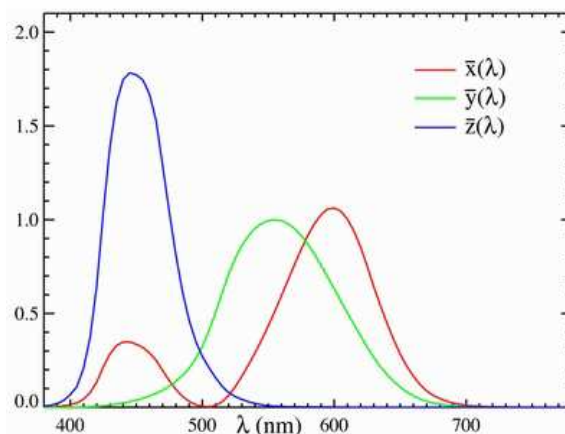


Fig.1.1 – Curve di risposta standard dell'occhio [3]

La radiazione visibile all'occhio umano in realtà, risulta essere una porzione sensibilmente ridotta dello spettro di radiazioni elettromagnetiche, le quali sono disposte in una porzione ampiamente superiore a quella percepibile dall'occhio, come è possibile notare in Fig. 1.1.2:

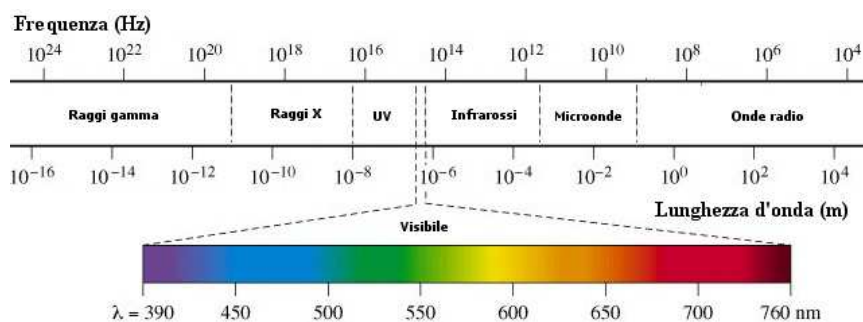


Fig. 1.1.2 – Spettro di radiazioni elettromagnetiche [4]

Dalla Fig.1.1.2, si nota anche la differente lunghezza d'onda di appartenenza di ogni colore in relazione all'energia necessaria richiesta secondo il principio della *Legge di Plank*. Il principio esprime la relazione tra energia emessa da un fotone in una radiazione elettromagnetica e la frequenza, dipendente dalla lunghezza d'onda secondo il rapporto  $\lambda = c/f$ .

La legge di Plank espressa dalla relazione  $E = h * f$ , mette in evidenza come all'aumentare della lunghezza d'onda diminuisce l'energia contenuta nella radiazione e viceversa.

## 1.2 – Grandezze fotometriche fondamentali [4]

Considerando sorgenti luminose di diversa natura e tipologia, bisogna considerare anche il sistema di misura utilizzato per rilevare i parametri e quantificarli correttamente. Nel caso dell'illuminotecnica e fotometria, il *Sistema Internazionale (SI)* definisce apposite unità di

misura da utilizzare in questo settore. Le *grandezze fotometriche* ( $X_v$ ) sono definite in relazione alla *grandezza radiometrica* ( $X_e$ ) pesata con la *curva di sensibilità fotopica* dell'occhio umano ( $V(\lambda)$ ), secondo la relazione riportata:

$$X_v = K_m * \int_{\lambda_{min}}^{\lambda_{max}} X_e(\lambda) * V(\lambda) d\lambda$$

dove con  $K_m = 683$  [Lumen/Watt] si indica un fattore costante che permette di avere un legame diretto con l'unità di misura *dell'Intensità Luminosa*, espresso nel S.I. in *Candele* [cd].

### 1.2.1 – Flusso luminoso

Il *Flusso Luminoso* è definito come la quantità di energia irradiata in ogni direzione da una sorgente luminosa nello spazio.

$$X_v = K_m \cdot \int_{\lambda_{min}}^{\lambda_{max}} P(\lambda) \cdot V(\lambda) d\lambda$$

Esso è individuato dall'unità di misura *LUMEN* [Lm] indicato con il simbolo [  $\phi$  ]. Il lumen è identificato come il flusso luminoso emesso nell'angolo solido unitario, da una sorgente puntiforme, posta al centro di una sfera d'intensità luminosa pari a una candela [cd] in tutte le direzioni. Per rappresentare la grandezza dell'angolo solido, il S.I. identifica lo *STERADIANTE* [sr] da cui si ottiene che 1 lumen = 1cd x sr .

Per compiere misure di flussi luminosi, è necessario utilizzare strumentazioni in grado di captare il flusso emesso in tutte le direzioni dalla sorgente presa in esame. La misura deve essere eseguita al chiuso in condizioni controllate e con uno strumento in grado di diffondere il più possibile il flusso emesso cercando di limitare quanto più è possibile il disperdersi per assorbimento della radiazione con lo strumento stesso. Lo strumento utilizzato è la *SFERA DI ULBRICHT* o *SFERA INTEGRATRICE* riportata in Fig.1.2, composta da una superficie di materiale plastico o in fibra di vetro e resina, con la superficie interna tinta con una vernice bianca avente un coefficiente di riflessione pressoché unitario per permettere la massima riflessione in tutte le direzioni. All'interno della sfera, sono disposti i sensori elettronici che compiono la misura, separati dalla sorgente presa in esame da un setto separatore, per impedire la misura diretta del flusso, e quindi ottenere una misura più uniforme e globale. La sorgente in prova è inserita nel centro della sfera e mantenuta in tale posizione da un supporto metallico. Inoltre la sfera è dotata anche di una sorgente luminosa ausiliaria di riferimento di cui si conoscono i parametri illuminotecnici come intensità luminosa, flusso e illuminamento, con il quale è effettuato il confronto e la calibrazione della sorgente in prova.

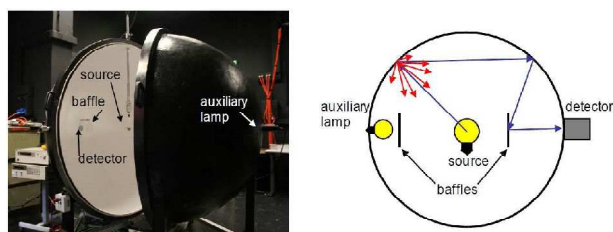


Fig. 1.2 – Sfera integratrice [5]

Essendo noti i parametri illuminotecnici della sorgente di riferimento ed essendoci proporzionalità tra il flusso emesso e l'illuminamento, si può ottenere il flusso emesso dalla sorgente presa come prova attraverso le seguenti espressioni:

$$E_t = K * \varphi_t \quad E_r = K * \varphi_r \quad \varphi_t = (E_t/E_r) * \varphi_r$$

### 1.2.2 – Luminanza

La *Luminanza* è uno dei parametri principali e più importanti tra quelli riguardanti il settore illuminotecnico e fotometrico. Essa è definita come il rapporto tra l'intensità proveniente da una superficie luminosa nella direzione di osservazione e l'area della proiezione ortogonale di quella superficie sul piano ortogonale a questa direzione Fig. 1.2.2.

La luminanza viene indicata con la lettera [ L ] e la sua unità di misura è la candela al metro quadro [cd/mq].

$$L = \frac{dI}{dA_{app}} = \frac{d\phi^2}{d\Omega(dA * \cos\theta)} \text{ [cd/m}^2\text{]}$$



Fig.1.2.2 – Superficie luminosa [6]

Le superfici luminose secondarie o riflettenti, sono spesso indicate come superfici lambertiane, cioè garantiscono la diminuzione dell'intensità luminosa all'aumentare dell'angolo di osservazione ( $\theta$ ), mantenendo dunque la luminanza costante ed eliminando di fatto la dipendenza della luminanza dall'angolo di osservazione. Oltre a questi fattori, la luminanza ha due proprietà fondamentali:

- La luminanza è indipendente dalla distanza;
- La luminanza è una quantità che si conserva nel sistema;

Dall'ultima proprietà si nota come la conservazione di luminanza è importante anche riguardo alla porzione interna dell'occhio umano, il quale non altera il livello di luminanza in funzione della densità di flusso ricevuto, e di fatto rappresenta quella che viene comunemente indicata come Luminosità.

Per fare misure di Luminanza è utilizzato un misuratore denominato *LUMINANZOMETRO*. La luminanza di una superficie dopo essere stata convogliata attraverso appositi passaggi

costituiti da lenti e filtri fotopici su una superficie fotosensibile, viene misurata a partire dalla conversione in illuminamento, ottenendo una corrente proporzionale alla luminanza stessa come riportato in Fig. 1.2.3 Questo approccio non consente di ottenere una misura della radianza spettrale, e quindi conoscere le lunghezze d'onda della sorgente luminosa scomposta nel suo spettro fondamentale.

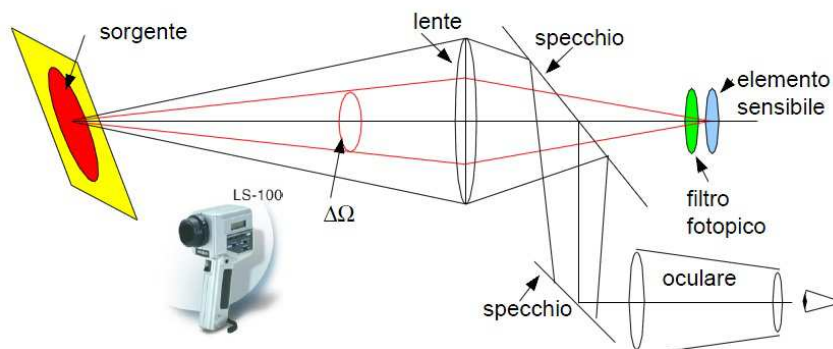


Fig.1.2.3 – Luminanzometro[5]

Per compiere la misura di radianza spettrale, è utilizzato lo *SPETTORADIOMETRO*, in grado di analizzare lo spettro di una sorgente che convoglia attraverso una fenditura ricavata in una lente separatrice, una porzione di radiazione luminosa inviandola verso un monocromatore, in grado di scomporre la radiazione nei suoi elementi fondamentali, trasferendole una a una tramite un selettore al fotorilevatore dello spettroradiometro. Di seguito in Fig.1.2.4 è riportato lo schema esplicativo del funzionamento descritto:

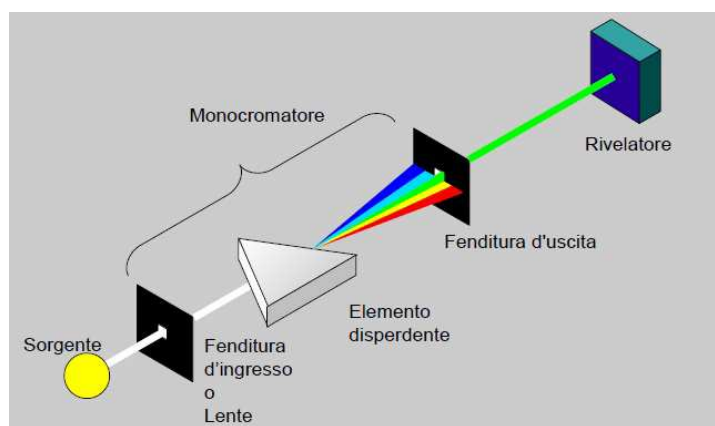


Fig.1.2.4 - Spettroradiometro

### 1.2.3 – Illuminamento

*L'illuminamento* è la terza grandezza del settore illuminotecnico e fotometrico presa in considerazione per lo sviluppo di questo lavoro di tesi. Esso è definito come il rapporto tra il flusso luminoso emesso da una sorgente verso una determinata superficie, e l'area della superficie trattata, oggetto dello studio illuminotecnico. L'illuminamento è indicato con la

lettera E, la sua unità di misura è il *Lux* [lx]. Di seguito è riportato il rapporto nei termini espressi in precedenza.

$$E = \frac{d\Phi}{dA}$$

Lo studio d'illuminamento non si discosta molto dalle precedenti grandezze fotometriche esaminate in precedenza, poiché collegato direttamente alla luminanza, tramite un coefficiente di luminanza indicato con la lettera [ q ].

$$q = \frac{L}{E} [sr^{-1}]$$

Il coefficiente di luminanza varia il suo valore in funzione della tipologia di sorgente che viene assunta durante il calcolo. Infatti, se la superficie presa in esame viene considerata Lambertiana, come il caso della sfera integratrice, il coefficiente di luminanza varia secondo la relazione riportata di seguito:

$$q = \frac{\rho}{\pi} \qquad \rho = \frac{\Phi_r}{\Phi_i}$$

dove con il termine  $\rho$  viene indicato il *coefficiente di riflessione* come rapporto tra *flusso luminoso riflesso* dalla superficie ( $\Phi_r$ ) e il *flusso luminoso incidente* ( $\Phi_i$ ) sulla superficie.

Dunque per una superficie lambertiana diffondente, illuminamento e luminanza, sono direttamente collegati, secondo le relazioni:

$$L = \frac{\Delta \Phi_r}{\pi \Delta A} \quad \rho = \frac{\Phi_r}{\Phi_i} \quad \Rightarrow \quad L = \frac{\rho \Delta \Phi_i}{\pi \Delta A}$$

$$E = \frac{\Delta \Phi_i}{\Delta A} \qquad q = \frac{L}{E} = \frac{\frac{\rho \Delta \Phi_i}{\pi \Delta A}}{\frac{\Delta \Phi_i}{\Delta A}}$$

$$q = \frac{\rho}{\pi}$$

$$L = \frac{\rho}{\pi} E$$

Per eseguire misure d'illuminamento, lo strumento utilizzato è il *LUXMETRO*, composto di un nucleo rilevatore con suo interno un fotodiodo al silicio. Il fotodiodo composto di uno



strato P e da uno strato N, genera una corrente proporzionale all'intensità di illuminamento ricevuto dal sensore di misura. Questa tipologia di strumentazione mantiene a lungo la calibrazione e lavora in maniera ottimale secondo un funzionamento in corto circuito, in modo da effettuare misure di corrente senza subire interazioni o alterazioni da agenti esterni. Prima di procedere alla misura, il luxmetro deve essere tarato e calibrato in funzione dell'illuminamento che si andrà a misurare, in modo da effettuare una lettura tale da poter essere considerata qualitativamente attendibile, nell'intorno della curva di sensibilità fotopica. Attraverso l'ausilio di una sorgente ad incandescenza di cui si conoscono tutti i parametri elettrici e illuminotecnici, è possibile tarare lo strumento di misura, mentre per effettuare le calibrazioni della sonda in funzione dell'illuminamento da misurare, vengono inseriti speciali filtri nella cavità contenente il fotodiodo. Di seguito sono riportati in Fig.1.2.5 un luxmetro ed il particolare del fotodiodo[7]:



Fig.1.2.5 – Luxmetro[8]

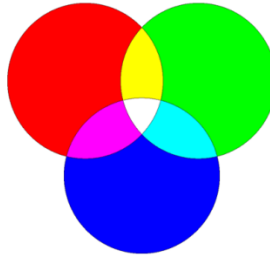
### 1.3 – Colorimetria

Dallo studio dei percettori principalmente utilizzati nell'occhio umano e lo studio dei modelli matematici utili per quantificare le grandezze illuminotecniche necessarie, si nota come una determinata cromaticità o colore è dipendente dalla somma additiva di tre differenti colori alle diverse lunghezze d'onda individuate, grazie ai differenti coni fotorecettori, in un unico colore. Il colore quindi può essere assunto come un punto definito all'interno di uno spazio tridimensionale, le cui coordinate sono legate alla quantità di colori necessari per riprodurlo.

#### 1.3.1 – Leggi di Grassman

Alla base della colorimetria sono utili e fondamentali tre leggi individuate da GRASSMAN:

1. Tre grandezze specificano il colore;
2. La sensazione di colore è proporzionale allo stimolo;
3. Una mescolanza additiva di colori dipende solo dal loro aspetto.



Queste leggi sono state ridefinite da HUNT:

1. Una sensazione di colore è completamente specificata da tre grandezze: la tinta, l'intensità del colore, l'intensità del bianco;
2. Se una luce varia con continuità, anche la sensazione di colore della mescolanza additiva con una seconda luce fissata varia con continuità;
3. Il risultato di una mescolanza additiva di colori dipende solo dal loro aspetto e non dalla loro composizione fisica (spettro).

#### 1.4 – La tecnologia Light Emitting Diode (LED)

La tecnologia LED ha ottenuto un notevole balzo in avanti nell'ultimo decennio, durante la corsa al miglioramento dell'efficienza luminosa e al risparmio energetico, tanto da far affermare i LED in una buona porzione del mercato dell'industria illuminotecnica, surclassando di gran lunga le ormai obsolete sorgenti alogene. Oltre la ridotta grandezza della giunzione p-n di cui è composto il diodo, la sua affermazione nelle diverse realtà industriali e civili è dovuta anche alla grande quantità di ore di funzionamento ininterrotte, combinato con il risparmio energetico ottenuto. In Fig.1.4 è riportato il modello costruttivo di un LED nella sua forma e dimensione più semplice:

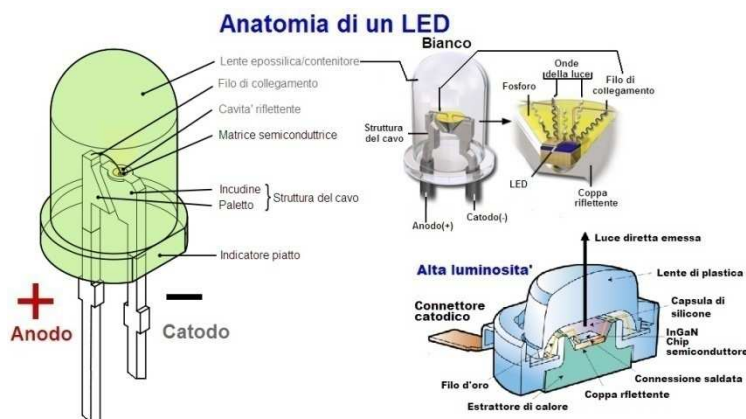


Fig.1.4 – Modello costruttivo di un LED[9]

Il funzionamento di un diodo LED è basato sul principio della ricombinazione tra un elettrone proveniente dalla zona n con una lacuna lasciata libera nella zona polarizzata positivamente. Quando la giunzione è polarizzata direttamente, si ha la migrazione di una cospicua quantità di elettroni dalla banda di conduzione alla banda di valenza. Questa ricombinazione avviene attraverso il rilascio di una certa quantità di energia sotto forma di

fotone, la cui cromaticità dipende dalla tipologia di materiale utilizzato per la costruzione della giunzione stessa.

I diodi LED oltre ad avere un'alta efficienza luminosa (80 – 100 lm/W) offrono anche la possibilità di essere utilizzati e comandati attraverso un controllo sulla corrente erogata, producendo colori saturi, con una ridotta quantità di componenti ultravioletti e infrarossi presenti nello spettro di emissione. In Fig.1.4.2, è riportato un confronto tra lo spettro di luce solare, completa di tutte le componenti alle diverse lunghezze d'onda, e le lampade comunemente utilizzate tra cui i LED.

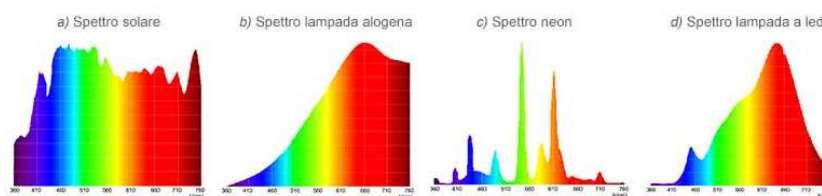


Fig.1.4.2 – Confronto tra spettri di emissione [10]

Come si può notare, la lampada alogena manca di frequenze fredde, il neon è alquanto selettivo mentre il LED in modo quasi omogeneo ricopre una buona quantità dello spettro. Per la stesura di questo lavoro di tesi, sono stati utilizzati LED ad alta potenza, per permettere di ottenere la corretta cromaticità richiesta dallo studio in questione. Un esempio di LED di potenza è riportato in Fig.1.4.3 con il particolare della zona di svuotamento:

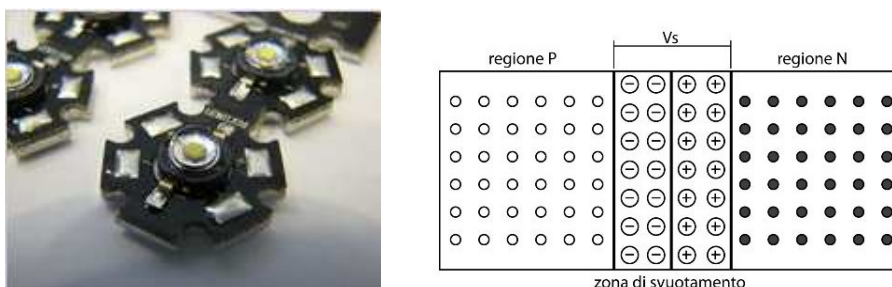


Fig. 1.4.3 – LED di potenza (a) [9] – Zona di svuotamento (b)[11]

### 1.5 – Modulazione della larghezza d'impulso (PWM)

Il microcontrollore di Arduino controlla e invia i comandi di attuazione ai vari elementi installati sul circuito elettrico in esame, attraverso le uscite denominate PWM [12]. La tecnica *Pulse Width Modulation (PWM)*, è una tipologia di modulazione digitale utilizzata nella conversione dc-dc, grazie alla quale è possibile ottenere una tensione dal valore medio desiderato, direttamente proporzionale al rapporto tra la *durata dell'impulso positivo*(ton), e di quello negativo (DUTY CYCLE), ottenuto come il periodo totale della commutazione ( $T_s$ ) al quale è stato sottratto il periodo dell'impulso positivo. In Fig.1.5 è riportato un interruttore che simula la commutazione dal valore massimo al valore medio desiderato in uscita, in funzione della frequenza di commutazione dell'interruttore, mentre nel grafico è

riportata la caratteristica della tensione di uscita modulata ed applicata ad un carico puramente resistivo.

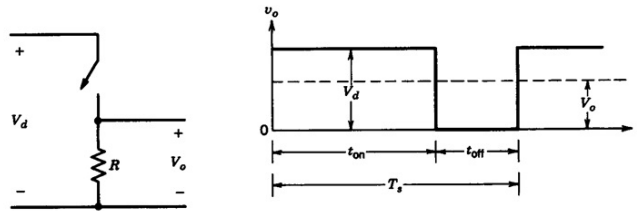


Fig.1.5 – Interruttore (a) e Caratteristica dell'onda di uscita modulata(b)

Avendo utilizzato componenti elettrici quasi interamente di tipo passivo, questo ci permette di prendere in esame la caratteristica in Fig.1.5 corrispondente ad una tensione media variabile regolata e applicata ad un carico resistivo. Si nota che applicando al carico resistivo una tensione in ingresso continua del valore \$V\_d\$, si può ottenere la tensione media in uscita calcolata, pari a \$V\_o\$ in relazione al duty cycle:

$$V_o = \frac{1}{T_s} * \int v_o(t) dt = \frac{1}{T_s} * \left( \int_0^{t_{on}} V_d dt + \int_{t_{on}}^{T_s} 0 dt \right) = \frac{t_{on}}{T_s} * V_d = D * V_d$$

Per migliorare ulteriormente la bontà della forma d'onda della tensione d'uscita, è possibile introdurre all'interno del circuito un comune filtro RC Passa Basso in modo da filtrare le componenti armoniche di tensione alle alte frequenze, ed ottenere un ulteriore miglioramento della forma d'onda continua della tensione in uscita \$V\_o\$.

### 1.6 – Sistema di controllo

Quando si opera con un sistema controllato, l'obiettivo principale è quello di ottenere un controllore \$C(s)\$ in un sistema dotato di retroazione che risulti BIBO stabile. [13] Un sistema *BIBO* presenta un segnale in ingresso limitato al quale corrisponde un segnale in uscita anch'esso limitato nel rispetto di requisiti fondamentali. Per operare il controllo, si passa dal dominio del tempo al dominio delle Laplace trasformate attraverso la funzione \$G(s)\$.

Lo schema esplicativo di un sistema controllato dotato di retroazione è riportato in Fig.1.6:

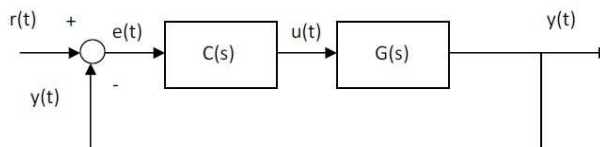


Fig.1.6 – Sistema controllato con retroazione

In automazione la retroazione permette ai sistemi dinamici di verificare la grandezza in uscita dal sistema, e in funzione del risultato ottenuto, andare a modificare il valore del parametro fornito in ingresso, secondo le necessità richieste. Infatti in un controllo in

retroazione il valore della variabile erogata all'uscita del sistema, viene letta ed elaborata dal controllore che opera un'opportuna regolazione dei parametri in ingresso al sistema, per ottenere in uscita un valore del risultato atteso migliorato. Proprio questa caratteristica ci permette di distinguere i sistemi retroazionati ad anello, da quelli in catena aperta sprovvista di retroazione.

Il sistema retroazionato presenta una funzione di trasferimento pari a:

$$W(s) = \frac{C(s) * G(s)}{1 + C(s) * G(s)}$$

Per effettuare ed implementare un controllo adeguato al circuito in esame, si è scelto di adoperare un *regolatore PID*, cioè un regolatore in grado di compiere tre azioni distinte: *Proporzionale, Integrale e Derivativa*. Esso esegue un controllo sull'uscita del sistema, leggendo il valore ottenuto in retroazione, e andando a calcolare il valore da assegnare ai parametri in ingresso affinché l'errore tra ingresso e uscita risulti nullo. Le tre azioni di controllo eseguite dal controllore sono:

- *PROPORZIONALE*: il valore del segnale d'errore è calcolato come differenza tra il segnale in ingresso ed una grandezza presa come riferimento;
- *INTEGRALE*: tiene conto dei valori passati del segnale di errore;
- *DERIVATIVA*: tiene conto della variazione di velocità del segnale di errore.

In Fig.1.6.2 è riportato lo schema di un regolatore PID:

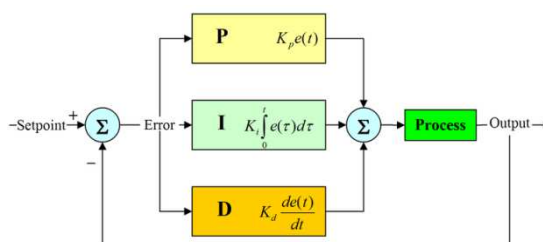


Fig.1.6.2 – Regolatore PID

In Fig. 1.6.2 si notano i tre blocchi operanti ognuno una funzione differente. Inoltre si nota come le azioni di regolazione confluiscono tutte in un blocco sommatore, il quale esegue la somma algebrica dei diversi contributi per fornire in uscita al sistema, il valore complessivo della regolazione effettuata, secondo la relazione:

$$u = u_p + u_I + u_D$$

Il primo membro descrive l'azione proporzionale ottenuta moltiplicando il segnale d'errore per la *costante moltiplicativa proporzionale* ( $K_p$ ):

$$u_p = k_p * errore$$

Anche se un regolatore può avere solo l'azione proporzionale, si predilige l'introduzione delle altre due azioni di controllo per evitare a regime un'azione nulla di comando sull'errore.

Il secondo termine dell'equazione rappresenta l'azione integrale, proporzionale all'integrale nel tempo del segnale di errore:

$$u_p = k_i * \int errore(t) dt$$

Grazie a questa tipologia d'azione, il regolatore è in grado di fornire un'uscita anche quando l'errore è stato annullato.

Il terzo e ultimo elemento dell'equazione, esprime il contributo derivativo della regolazione PID. Esso è definito come la derivata nel tempo dell'errore moltiplicato per la *costante di derivazione (Kd)*:

$$u_D = k_D * \frac{\partial errore}{\partial t}$$

L'azione derivativa tende a migliorare le prestazioni del regolatore, andando a compensare le variazioni del segnale d'errore in funzione delle variazioni di velocità che il segnale stesso assume.

## *CAPITOLO 2 - VISIONE DEL SISTEMA ELETTRICO PREESISTENTE*

### *2.1 – Hardware – Descrizione del sistema elettrico di partenza*

Come già anticipato nel paragrafo introduttivo, questo lavoro di tesi inizia il suo percorso sulla base di un progetto illuminotecnico già esistente, in grado di adempiere a determinate funzioni di misura e impostazioni di parametri elettrici ed illuminotecnici. Attraverso un sistema costituito da tre schede circuitali, è possibile comandare una sorgente luminosa composta da un numero predefinito di LED. La prima scheda è adibita al controllo e gestione dell'alimentazione, attraverso appositi alimentatori in grado di controllare la corrente erogata. La seconda scheda permette di monitorare e controllare il sistema di frequenze generate dalla radiazione luminosa, attraverso il controllo e la gestione di appositi sensori per il rilevamento delle frequenze. La terza scheda invece, ultima a essere inserita nel sistema complessivo, realizza il controllo di temperatura del sistema contenente i LED, consentendo il funzionamento in condizioni termiche definite, attraverso l'ausilio di riscaldatori e una ventola per il raffreddamento.

Di seguito sono descritti e analizzati singolarmente i differenti componenti del sistema preesistente.

#### *2.1.1 – Sorgenti luminose impiegate*

Il progetto è basato sulla sintesi additiva di differenti colori secondo i principi descritti dalle Leggi di Grassman viste in precedenza. Per questo motivo è intuibile come i LED da prendere in esame, devono rispondere a determinati parametri per far sì che il progetto possa essere attendibile e stabile. Il numero di LED e i colori scelti per questo progetto sono:

- N° 2 Led Rossi da 55 lm ciascuno per un totale di 700 mA assorbiti;
- N° 2 Led Verdi da 55 lm ciascuno per un totale di 700 mA assorbiti;
- N° 5 Led Blu da 20 lm ciascuno per un totale di 700 mA assorbiti;

Dopo attente considerazioni sui LED da impiegare, si è giunti alla conclusione di utilizzare i *LUXEON III STAR* prodotti dalla *LUMILEDS*. In Fig.2.1.1 sono riportati i modelli descrittivi con i parametri fondamentali e utili dei LED:

LUMILEDS

Luxeon III Star

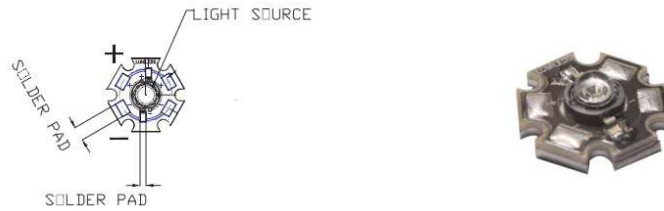


Fig.2.1.1 – LED utilizzati

In Tab.2.1.1 sono riportate le caratteristiche luminose a 700 mA ad un Temperatura di Giunzione pari a  $T_j = 25^\circ\text{C}$

Color	Luxeon Emitter	Minimum Luminous Flux ( $\text{lm}$ ) or Radiometric Power ( $\text{mW}$ ) $\Phi_V^{1-2}$	Typical Luminous Flux ( $\text{lm}$ ) or Radiometric Power ( $\text{mW}$ ) $\Phi_V^{1-2}$	Radiation Pattern
White	LXHL-LW3C	60.0	65	Lambertian
Green	LXHL-LM3C	51.7	64	
Cyan	LXHL-LE3C	51.7	64	
Blue <sup>®</sup>	LXHL-LB3C	13.9	23	
Royal Blue <sup>®</sup>	LXHL-LR3C	275 mW	340 mW	
White	LXHL-FW3C	51.7	58	Side Emitting
Green	LXHL-FM3C	51.7	58	
Blue <sup>®</sup>	LXHL-FB3C	13.9	21	

Tab.2.1.1 – Caratteristiche luminose del LED

Dopo aver scelto il numero di LED da utilizzare e il metodo di collegamento tra di essi, è stata studiata la migliore disposizione sul supporto, in modo da permettere il facile raggiungimento dei pin di collegamento per l'alimentazione e permettere contemporaneamente il necessario scambio termico richiesto per il raffreddamento. Infatti, se i LED fossero stati installati su supporti inappropriati senza tener conto del necessario coefficiente di scambio termico, la temperatura raggiunta dalla giunzione, avrebbe potuto causare il decadimento delle caratteristiche illuminotecniche e quindi delle prestazioni desiderate.

La scelta del supporto è stata indirizzata verso una piastra metallica in alluminio di forma quadrata con dimensione del lato pari a 11.5 cm e spessore 1 cm, con una superficie totale di  $132.25 \text{ cm}^2$ , sulla quale sono stati installati mediante perni metallici i LED, secondo una disposizione radiale, inscritti in una circonferenza. Attraverso i fori filettati denominati M3 sono stati fissati i LED sul supporto circolare utilizzato come base e tramite i fori filettati denominati M4, è stata fissata la base circolare al supporto principale. Tramite l'utilizzo di una pasta ad alta conducibilità termica inserita tra il supporto circolare metallico e la piastra quadrata, si garantisce un efficiente livello di scambio termico. In Fig.2.1.2 è riportato lo schema con la disposizione dei LED sul supporto circolare e sul supporto quadrato:



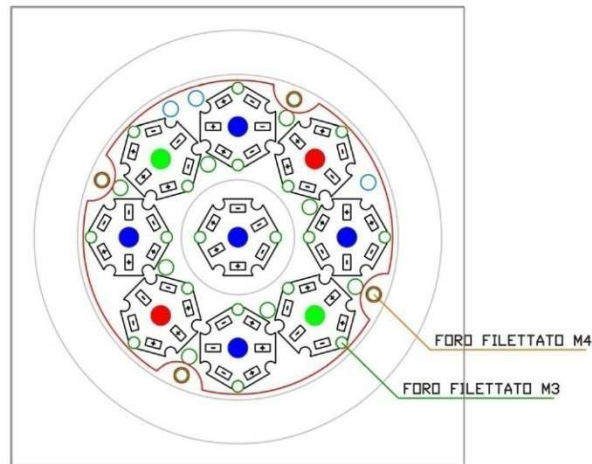


Fig.2.1.2 – Disposizione dei LED sul supporto circolare [5]

Dopo aver installato i LED, il supporto metallico è stato collegato opportunamente ad una sfera integratrice miniaturizzata, per permettere il contenimento della radiazione luminosa ed impedire che radiazioni luminose non desiderate o casuali entrino nella sfera e influiscano negativamente durante il processo di misura, discostando dal valore atteso i risultati ottenuti.

### 2.1.2 – Sfera Integratrice

La sfera integratrice miniaturizzata utile per contenere il flusso luminoso emesso dalle sorgenti, è stata realizzata attraverso la stampa 3D con un materiale semi-plastico in grado di fornire alla sfera sia integrità e robustezza ma anche leggerezza. La sfera in esame presenta una tinta di colore nero sulla superficie complessiva esterna, mentre le pareti interne sono rivestite di una speciale tinta bianca, avente proprietà riflessive maggiorate. Inoltre all'interno è presente un setto separatore utilizzato per fornire una radiazione emessa non diretta sullo strumento di misura, ma ottenuta tramite multiple riflessioni della radiazione emanata sulle pareti interne del sistema, in modo da ottenere una misura più omogenea. Il setto separatore ha anche lo scopo di proteggere dalla radiazione diretta l'utente che si accinge a osservare attraverso un'apposita apertura, la tipologia di misura in atto. Una seconda apertura circolare è stata ricavata al di sopra del foro per l'osservazione, ed è stata utilizzata per alloggiare i sensori per la misura di frequenza a partire dalla radiazione luminosa richiesta dall'utente. In Fig.2.1.3 è riportata la sfera integratrice utilizzata:



*Fig.2.1.3 – Sfera integratrice [5]*

Nella zona posteriore della sfera, si nota la presenza della piastra metallica in alluminio di colore argentato, e si nota inoltre la presenza dei conduttori di collegamento per l'alimentazione dei LED, del sensore di temperatura, della ventola e dei riscaldatori. Nella zona frontale invece si nota l'apertura circolare per permettere l'osservazione della sintesi additiva ottenuta, mentre al di sopra si trova la seconda apertura nella quale sono posizionati i sensori per la misura della frequenza.

## 2.2 – Circuito di alimentazione dei LED

Uno dei componenti più importanti del sistema è costituito dal circuito di alimentazione dei LED. Il circuito infatti controlla ed eroga la corrente attraverso un sistema di alimentatori controllati tramite un determinato valore di tensione fornito in ingresso al pin di controllo. In Fig.2.2 è riportato il circuito utilizzato in precedenza contenente i componenti elettronici necessari per il corretto funzionamento e la gestione dei parametri elettrici ai LED:



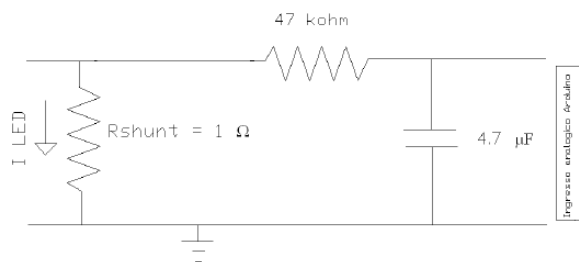
*Fig.2.2 – Circuito di alimentazione dei LED [5]*

Nel circuito sono visibili le otto morsettiere di colore verde utilizzate per inviare alle serie di LED la corrente e la tensione necessaria al loro funzionamento. Sei morsettiere sono suddivise in funzione del colore dei LED, tre forniscono corrente e tensione mentre le altre

tre, poste lateralmente a quelle di alimentazione, costituiscono i punti di riferimento per permettere le misurazioni dei parametri elettrici. Due morsettiere, costituiscono i punti di arrivo della potenza necessaria al circuito ed ai LED per il funzionamento. Sono necessarie due morsettiere distinte per l'alimentazione esterna poiché i diversi gruppi di LED sono alimentati con differenti valori di tensione. La serie composta dai due LED di colore rosso e quella composta dai due LED di colore verde, sono alimentate da una tensione continua di 12 V, mentre la serie dei cinque LED blu è alimentata da una tensione continua di 24 V.

### 2.2.1 – Misura di corrente inviata ai LED

I tre parallelepipedi verticali di colore bianco situati nelle vicinanze delle morsettiere di alimentazione dei LED, costituiscono i resistori di shunt da 1Ω utilizzati per effettuare la misura di corrente erogata ai LED secondo lo schema riportato di seguito:



La corrente erogata, per poter essere elaborata dal sistema a microcontrollore Arduino, deve essere trasformata in un corrispondente valore di tensione. Tramite un filtro RC Passa Basso sono filtrate le armoniche di tensione, eliminando i disturbi alle alte frequenze che alterano lo spettro del segnale prima di essere inviato all'ingresso analogico di Arduino per essere elaborato. Infatti attraverso l'ausilio di una resistenza di filtro da 47 kΩ e un condensatore di filtro al tantalio di capacità 4.7 μF, si può ottenere la pulsazione di taglio che permette di eliminare i disturbi indesiderati alle frequenze:

$$f_t = \frac{1}{2 * \pi * R * C} = \frac{1}{2 * \pi * 47 * 10^3 * 4,7 * 10^{-6}} \sim 10 \text{ Hz}$$

### 2.2.2 – Alimentatori integrati dei LED

Gli elementi con maggior valenza presenti nel circuito di alimentazione dei LED, sono gli alimentatori dc-dc riportati in Fig.2.2.2, che permettono un'alimentazione con corrente impressa costante, proporzionale al valore di tensione di controllo  $V_{ctrl}$  inviata al pin dell'alimentatore denominato CTRL.

Gli alimentatori che soddisfano le nostre richieste sono i LUXDRIVE 3021 D-E-700 BuckPuck riportati in Fig.2.2.2:



Fig.2.2.2 – Alimentatore LuxDrive

Questa tipologia di alimentatori, permette di erogare e imprimere una corrente costante al gruppo di LED, secondo una tensione imposta al morsetto CTRL dell'alimentatore. La tensione  $V_{ctrl}$  può variare da un minimo di 0 V, al quale corrisponde il valore massimo di corrente lasciata passare, fino a 5 V, al quale corrisponde una corrente lasciata passare nulla. In Fig.2.2.3 è riportata la caratteristica degli alimentatori fornita dal costruttore che rende comprensibile il range di funzionamento e i valori ammissibili di  $V_{ctrl}$ . Il valore massimo di corrente lasciata passare in corrispondenza del valore minimo di  $V_{ctrl}$  è di 700 mA, corrispondente al valore massimo ammissibile per la tipologia di alimentatori scelti.

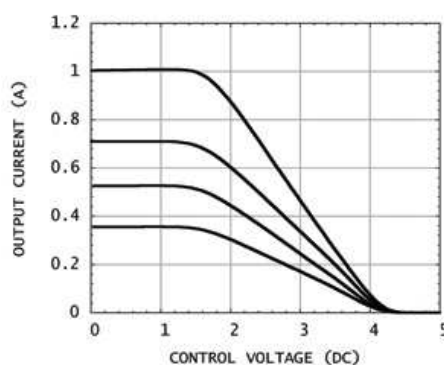


Fig.2.2.3 – Caratteristica fornita dal costruttore

Attraverso il datasheet degli alimentatori, si individua il metodo più opportuno da utilizzare per connettere i LED all'alimentatore. Infatti secondo lo schema riportato in Fig.2.2.4, si nota come la serie di LED viene collegata ai morsetti LED+ e LED- . La corrente e la tensione necessaria per far funzionare la serie di LED con valori specifici, è resa disponibile in ingresso all'alimentatore, tramite i due morsetti indicati con Vin+ e Vin- .

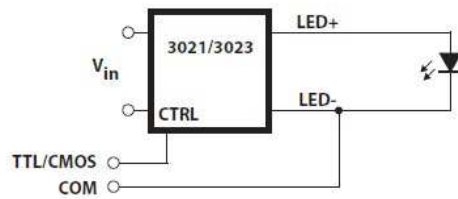


Fig.2.2.4 – Collegamenti dell'alimentatore

Per far sì che le fluttuazioni di corrente e i disturbi legati al ripple della tensione entranti nel sistema di alimentazione fossero ridotti il più possibile, è stato necessario introdurre condensatori in parallelo all'ingresso di ogni singolo alimentatore e all'ingresso generale del sistema di alimentazione. Infatti come riportato in Fig.2.2 si notano i 3 condensatori elettrolitici di colore blu da 220  $\mu\text{F}$  – 100V responsabili dell'attenuazione delle fluttuazioni della corrente, collegati in parallelo ad ogni uscita degli alimentatori; un condensatore elettrolitico di colore nero da 1000  $\mu\text{F}$  – 30V utilizzato come filtro per attenuare le fluttuazioni della tensione in ingresso generale al sistema, e tre condensatori ceramici da 100nF, 103 pF e 22 nF collegati in parallelo ad ogni singola alimentazione degli alimentatori. I tre condensatori ceramici riportati in Fig.2.2.5, sono situati nel vecchio circuito al di sotto della scheda elettronica stampata, poiché solo con calcoli successivi si è resa necessaria la loro introduzione.



Fig.2.2.5 – Condensatori ceramici

La tensione inviata al pin di controllo dell'alimentatore deve essere calcolata, misurata e controllata in ogni istante di funzionamento, e per questo motivo è stata effettuata una linearizzazione della caratteristica fornita dal costruttore, in modo da permettere la fornitura dei parametri elettrici individuati nella zona lineare della caratteristica. Conoscendo il valore massimo di corrente erogabile dagli alimentatori e il valore di tensione massima applicabile al pin di controllo per lo spegnimento totale dell'alimentatore stesso, si ottiene la caratteristica riportata in Fig.2.2.6:

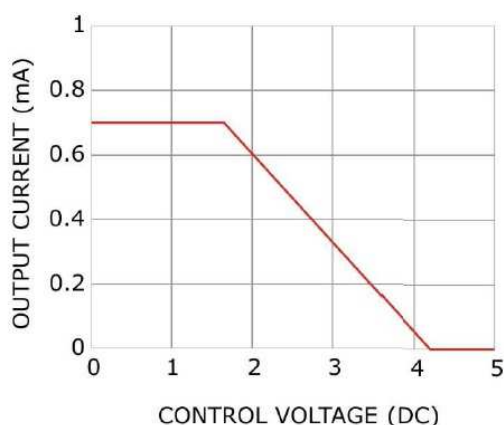


Fig.2.2.6 – Caratteristica dell'alimentatore implementata

Per questo motivo è possibile esprimere la relazione tra la corrente erogata dall'alimentatore e la tensione inviata al morsetto Vctrl come:

$$V_{ctrl} = -3,6 * I_{out} + 4,2$$

Nel capitolo successivo, in seguito alle modifiche apportate al circuito e al firmware di Arduino, la caratteristica e la relativa equazione, subiranno un cambiamento dovuto al nuovo circuito e all'inserimento dei nuovi componenti utilizzati.

### 2.2.3 – Erogazione della tensione di controllo al driver

Come accennato in precedenza, il valore di tensione fornito al morsetto di controllo dell'alimentatore, permette di modificare in funzione di essa, la corrente inviata al gruppo di LED. La tensione continua, è erogata dall'uscita PWM di Arduino ed opportunamente filtrata attraverso un filtro RC Passa Basso posto a valle dell'uscita, in modo da ridurre la frequenza di taglio ed ottenere un segnale quanto più continuo privo di disturbi dovuti alle alte frequenze. Eseguendo i calcoli, sono stati individuati i valori di resistenza e capacità che soddisfano le nostre richieste nell'intorno di  $R = 270 \Omega$  e  $C = 100 \mu F$ . Per il resistore è stata utilizzata una comune resistenza con potenza pari ad un quarto di Watt, mentre per il condensatore si è optato per uno al tantalio. In Fig.2.2.7 è riportato il circuito di collegamento del filtro RC sulla scheda:

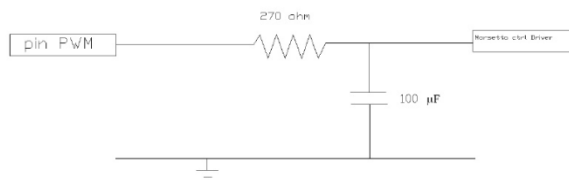


Fig.2.2.7 – Filtro RC

$$f_t = \frac{1}{2 * \pi * R * C} = \frac{1}{2 * \pi * 270 * 100 * 10^{-6}} \sim 6 \text{ Hz}$$

### 2.3 – Sensori per la misura della frequenza

Il sistema elettrico è dotato di un sensore per la rilevazione della frequenza a partire da una radiazione luminosa emanata dai LED durante il funzionamento. Il dispositivo che più soddisfa le nostre richieste è stato individuato nel TAOS TCS 230 riportato in Fig.2.3:



Fig.2.3 – Sensore TAOS

Il sensore scelto è composto da 64 fotodiodi disposti in una matrice 8x8, suddivisi in quattro gruppi da 16 elementi, ciascuno sottoposto ad un diverso filtro ottico individuato secondo i tre colori rosso, verde e blu ed un colore neutro. Il sensore ha degli ingressi comandati che permettono di abilitare funzioni come l'output (OE = out enable), selezionare la frequenza di fondo scala più vicina alle esigenze (S0,S1), oppure quali fotodiodi selezionare per effettuare la misura in funzione di un colore (S2,S3). L'uscita invece è un'onda quadra con duty cycle al 50% con una frequenza direttamente proporzionale all'illuminamento captato. In Fig.2.3.1 è riportato lo schema a blocchi di funzionamento del sensore:

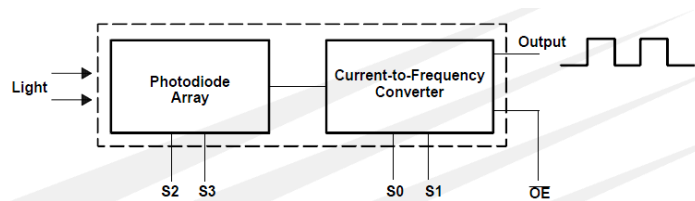


Fig.2.3.1 – Schema a blocchi del sensore

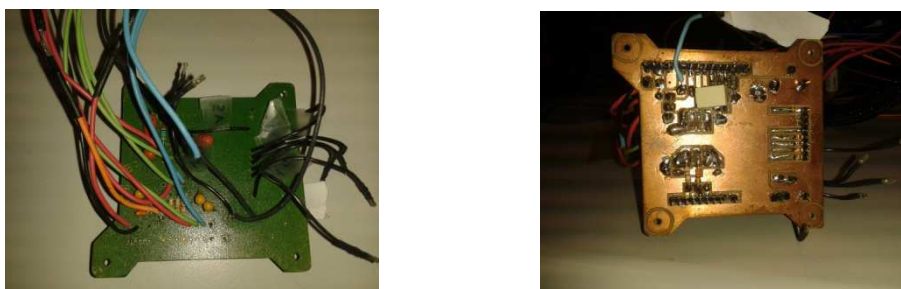
Nella Tab.2.3.1 sono riportati i valori che devono assumere le coppie d'ingressi e uscite del sensore, affinché si possa selezionare il fondo scala adeguato e il gruppo di fotodiodi opportuni:

Full-scale frequency (See Note 2)	S0 = H, S1 = H	500	kHz
	S0 = H, S1 = L	100	kHz
	S0 = L, S1 = H	10	kHz

S2	S3	PHOTODIODE TYPE
L	L	Red
L	H	Blue
H	L	Clear (no filter)
H	H	Green

*Tab.2.3.1 – Valori degli ingressi*

Il vecchio sistema adoperato per interfacciare i sensori TAOS con il microcontrollore Arduino, è composto da una scheda in rame sagomata contenente i punti di collegamento dei conduttori provenienti dai sensori, ed i morsetti in ingresso ad Arduino per la misura della frequenza. Questo supporto costituisce il secondo delle tre schede stampate che formavano il sistema preesistente. In Fig.2.3.2 sono riportate due immagini raffiguranti il fronte della scheda con tutti i conduttori in arrivo, e il retro con la presenza dei punti di collegamento tra conduttori ed Arduino.



*Fig.2.3.2 – Scheda di collegamento dei conduttori*

Il sensore vero e proprio contenente la matrice dei 64 fotodiodi, è di forma rettangolare con dimensioni dei lati pari a 4,1 x 5,1 mm. Data la loro ridotta dimensione, con un solo pulviscolo di polvere, si possono interdire i fotodiodi. Per ovviare a tale incomodo, il sensore è stato alloggiato in un corretto contenitore adatto ad essere installato sulla sommità della sfera integratrice. In Fig.2.3.3 sono riportate le immagini relative al sensore, al contenitore utilizzato e la sua relativa posizione sulla sfera integratrice.



*Fig.2.3.4 – Sensore TAOS (a) – Componente di supporto (b)*





Fig.2.3.4 – Supporto montato sulla sfera integratrice

### 2.4 – Circuito per il controllo della temperatura

Ultima scheda circuitale ad essere inserita durante il susseguirsi degli sviluppi del progetto, risulta essere quella per il controllo e la gestione della temperatura del sistema. Il circuito mette in relazione i componenti elettrici con il sensore per la misura di temperatura, fornendo corrente ai riscaldatori se la temperatura raggiunta dal sistema risulta inferiore ad un limite imposto, oppure attivando la ventola per il raffreddamento se la temperatura supera un limite massimo indesiderato. Il tutto è regolato dal codice macchina che il microcontrollore Arduino utilizza per far collaborare i diversi elementi, affinché il controllo sia mantenuto e rispettato. In Fig. 2.4 è riportato lo schema circuitale della scheda per il controllo:

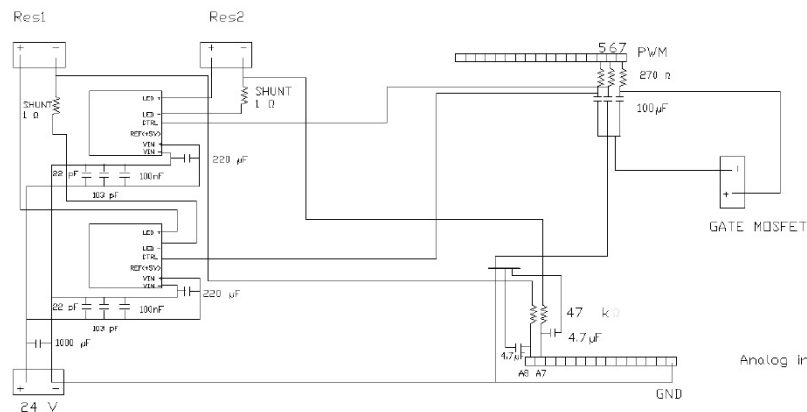


Fig.2.4 – Schema circuitale per il controllo di temperatura[14]

La componentistica utilizzata in questo circuito, è simile a quella utilizzata per la scheda di alimentazione dei LED. Infatti anche in questo caso sono presenti due alimentatori dc-dc che forniscono una corrente impressa costante, proporzionale al valore di tensione  $V_{ctrl}$  erogata all'alimentatore tramite l'uscita PWM di Arduino. Inoltre si nota la presenza dei resistori di shunt ceramici per la misura della corrente erogata ai LED. Le due morsettiere di colore verde, costituiscono i punti di collegamento dal quale i due riscaldatori montati sul retro della piastra in alluminio ricevono la potenza necessaria durante la fase di riscaldamento. Le altre due morsettiere invece, situate a sinistra e in basso in Fig.2.4.1, costituiscono

rispettivamente i punti di accesso per l'alimentazione della ventola e per l'alimentazione generale. Questo circuito utilizza una tensione nominale di funzionamento di 24 V continui.

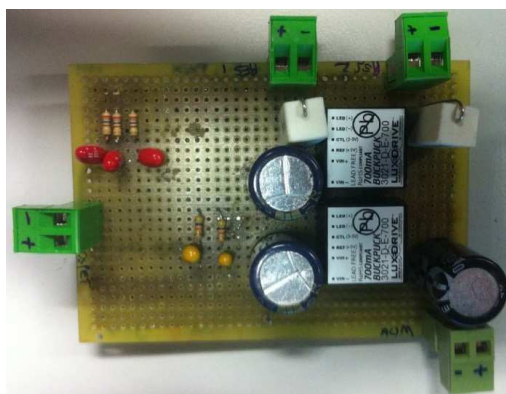


Fig.2.4.1 – Circuito elettrico del controllo di temperatura [14]

#### 2.4.1 – Sensore per la misura di temperatura del sistema

Per monitorare costantemente la temperatura del sistema composto dalla piastra e dal supporto LED, è stato introdotto un trasduttore di temperatura in una cavità ricavata sulla piastra in alluminio. Dopo attente osservazioni è stato usato per lo scopo il trasduttore della casa costruttrice *ANALOG DEVICES AD590* riportato in Fig.2.4.1.1:



Fig.2.4.1.1 – Trasduttore di temperatura

Il dispositivo è in grado di fornire in uscita una corrente continua proporzionale alla temperatura misurata secondo la relazione  $1 \mu A/K$  in conformità a quanto riportato nel datasheet dal costruttore. Esso è alimentato in ingresso da una tensione continua di 5V erogata da Arduino, e fornisce in uscita il relativo valore di temperatura misurata sotto forma di corrente. Essendo la temperatura ambiente mantenuta costante a circa  $25^{\circ}C$  all'interno del laboratorio durante tutto lo sviluppo del progetto, secondo quanto descritto in precedenza, la corrente continua in uscita risulta essere pari a circa  $298,2 \mu A/K$ . Purtroppo il dispositivo è collocato in un apposito alloggiamento interno alla piastra in alluminio e per questo non è stato possibile fotografare il suo reale aspetto. Di seguito in Fig.2.4.1.2 è riportato un estratto del datasheet fornito dal costruttore, con tutte le specifiche tecniche. Individuando la riga con il range contenente il valore di tensione imposto da Arduino, si nota che lo scostamento massimo e l'errore in  $\mu A$  della misura, risulta essere di  $0.5 \mu A/V$ .

### AD590J AND AD590K SPECIFICATIONS

Table 1. @ 25°C and  $V_s = 5\text{ V}$  unless otherwise noted

Parameter	AD590J			AD590K			Unit
	Min	Typ	Max	Min	Typ	Max	
<b>POWER SUPPLY</b>							
Operating Voltage Range	<b>4</b>		30	<b>4</b>		30	Volts
<b>OUTPUT</b>							
Nominal Current Output @ 25°C (298.2K)		298.2			298.2		$\mu\text{A}$
Nominal Temperature Coefficient		1			1		$\mu\text{A}/\text{K}$
Calibration Error @ 25°C			$\pm 5.0$			$\pm 2.5$	$^{\circ}\text{C}$
Absolute Error (over rated performance temperature range)							
Without External Calibration Adjustment			$\pm 10$			$\pm 5.5$	$^{\circ}\text{C}$
With 25°C Calibration Error Set to Zero			$\pm 3.0$			$\pm 2.0$	$^{\circ}\text{C}$
Nonlinearity							
For TO-52 and Flatpack packages			$\pm 1.5$			$\pm 0.8$	$^{\circ}\text{C}$
For 8-Lead SOIC package			$\pm 1.5$			$\pm 1.0$	$^{\circ}\text{C}$
Repeatability <sup>1</sup>			$\pm 0.1$			$\pm 0.1$	$^{\circ}\text{C}$
Long-Term Drift <sup>2</sup>			$\pm 0.1$			$\pm 0.1$	$^{\circ}\text{C}$
Current Noise		40			40		$\text{pA}/\sqrt{\text{Hz}}$
Power Supply Rejection							
$4\text{ V} \leq V_s \leq 5\text{ V}$		0.5			0.5		$\mu\text{A}/\text{V}$
$5\text{ V} \leq V_s \leq 15\text{ V}$		0.2			0.2		$\mu\text{A}/\text{V}$
$15\text{ V} \leq V_s \leq 30\text{ V}$		0.1			0.1		$\mu\text{A}/\text{V}$
Case Isolation to Either Lead		$10^{10}$			$10^{10}$		$\Omega$
Effective Shunt Capacitance		100			100		$\text{pF}$
Electrical Turn-On Time		20			20		$\mu\text{s}$
Reverse Bias Leakage Current <sup>3</sup> (Reverse Voltage = 10 V)		10			10		$\text{pA}$

<sup>1</sup> Maximum deviation between +25°C readings after temperature cycling between -55°C and +150°C, guaranteed, not tested.

<sup>2</sup> Conditions: constant 5 V, constant 125°C, guaranteed, not tested.

<sup>3</sup> Leakage current doubles every 10°C.

Specifications shown in **boldface** are tested on all production units at final electrical test. Results from those tests are used to calculate outgoing quality levels. All min and max specifications are guaranteed, although only those shown in **boldface** are tested on all production units.

Fig.2.4.1.2 – Datasheet fornito dal costruttore

Arduino non è in grado di elaborare un valore di corrente, per questo si opera una trasformazione da corrente in tensione, simile a quella adottata per il controllo di corrente inviata ai LED. Infatti la corrente proveniente dal trasduttore, viene indirizzata verso una resistenza di shunt da  $2,2\text{ k}\Omega$  la quale presenterà una caduta di tensione ai suoi capi proporzionale al valore di corrente in ingresso. Il valore di tensione misurata ai capi della resistenza, sarà inviato ad Arduino per compiere le dovute conversioni e tarature. Nel capitolo successivo, sarà evidenziata la differenza tra il valore di tensione ai capi della resistenza di shunt in relazione alla corrente ricevuta dal trasduttore e si implementerà il sistema in modo da ottenere la misura corretta del valore di temperatura. In Fig.2.4.1.3 è riportato lo schema circuitale contenente il principio di collegamento del sensore:

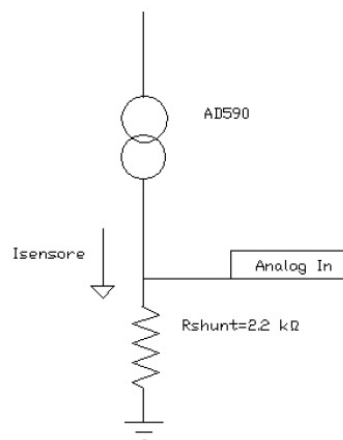


Fig.2.4.1.3 – Schema circuitali di collegamento del sensore[14]

### 2.4.2 – Riscaldatori del sistema

I resistori che più si avvicinano alle esigenze richieste dal sistema e dai calcoli, sono quelli in grado di fornire la stessa energia dissipata dai LED, nel rispetto dei vincoli circuitali e costruttivi. Le resistenze scelte per implementare questa funzione, sono le MPH20 della casa costruttrice TT ELECTRONICS riportate in Fig.2.4.2:



Fig.2.4.2 - Riscaldatore

Considerando la corrente massima che ogni singolo alimentatore è in grado di erogare, 700 mA, e considerando una tensione uguale a quella necessaria per alimentare i LED blu, la potenza diventa:

$$P = V * I = 3 V * 0.7 A = 2 W$$

Se si suppone che questa potenza da parte dei LED sia dispersa totalmente in calore, andando a considerare tutti i LED in funzione, si ottiene una potenza termica complessiva dissipata pari a:

$$P_{tot} = N_{led} * P = (2LED_r + 2LED_g + 5LED_b) * 2 = 20 W$$

dunque risulta intuibile che per dissipare questa potenza termica la resistenza utilizzata deve essere pari a :

$$R = \frac{P}{I^2} = \frac{20}{(0.7)^2} = 40.81 \Omega$$

con una tensione necessaria ai suoi capi di circa:

$$V = R * I = 40.81 \Omega * 0.7 A = 28.56 V$$

la quale risulta maggiore della tensione massima di 24 V da imporre al circuito.

In Fig.2.4.2.1 è riportata la caratteristica che individua la temperatura raggiungibile della flangia montata nella zona posteriore al resistore, in funzione della potenza dissipata, mentre in Fig.2.4.2.2 è riportato l'aumento di temperatura in funzione della potenza dissipata.

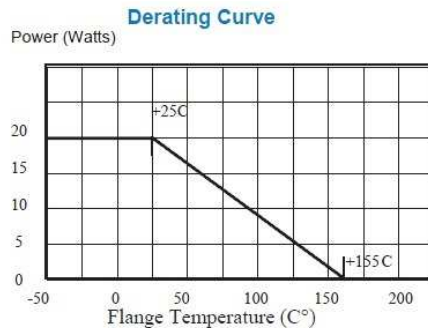


Fig.2.4.2.1 – Caratteristica di temperatura della flangia

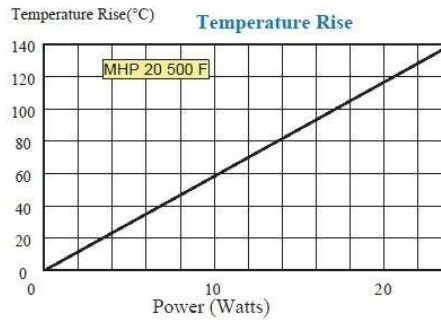


Fig.2.4.2.2 – Caratteristica di temperatura - potenza

Tuttavia, la potenza necessaria da smaltire, può essere ripartita attraverso l'utilizzo di due resistori distinti, ognuno con il suo alimentatore dc-dc e la sua caratteristica di funzionamento. In questo modo si ottiene una resistenza pari a:

$$R = \frac{P}{I^2} = \frac{20}{(0.7)^2} = 40.81 \Omega$$

Questo valore si discosta poco dai comuni resistori reperibili nei negozi di elettronica, tuttavia la presenza in laboratorio di resistori con un valore nominale di 50 Ω, ci consente di ottenere il valore di corrente da inviare al resistore in modo da avere una dissipazione nell'intorno di 20 W pari a:

$$I = \sqrt{\frac{P}{R}} = \sqrt{\frac{20}{50}} = 0.632 A$$

Con questa nuova corrente erogata ai resistori, si può ridurre la tensione utile ai suoi capi che risulta essere pari a:

$$V = R * I = 40 * 0.632 = 25.28 V$$

inferiore al valore di tensione individuato precedentemente.

In Fig.2.4.2.3 è riportato un particolare del datasheet fornito dal costruttore, con tutti i parametri termici ed elettrici necessari per capire il corretto funzionamento dei resistori:

## Specifications

Items	Specification			Conditions
Power Rating	20 Watts			-55 to 25 °C flange temperature
Power Rating	1.0 Watts			Free air.
Thermal Resistance	5.9 °CAW			Hot spot to Tab
Resistance Range	0.01-0.09 Ω	0.1-9.1 Ω	10-220 Ω	Up to 51 KΩ also available
Nominal Resistance Series	E6	E24	E24	2.5 Ω and 5.0 Ω also available
TCR	250 ppm/°C	100 ppm/°C	50 ppm/°C	-55 to +155 °C
Tolerance	+/- 5% (J)	5% (J)	+/-1% (F)	
Operation Temp. Range	-55°C to +155°C			
Max. Operating Volt.	500V or √ P.R			
Dielectric Withstanding Voltage	2000 Volts AC			60 seconds.
Load Life	ΔR +/- (1.0 % + 0.05 Ω)			25 °C, 90 min. ON, 30 min. OFF, 1000 hours.
Humidity	ΔR +/- (1.0 % + 0.05 Ω)			40°C, 90-95% RH, DC 0.1 W, 1000 hours.
Temp. Cycle	ΔR +/- (0.25 % + 0.05 Ω)			-55 °C, 30 min., +155 °C, 30 min., 5 cycles
Soldering Heat	ΔR +/- (0.1 % + 0.05 Ω)			250 +/- 5 °C, 3 seconds.
Solder ability	Over 95% of surface			230 +/- 5 °C, 3 seconds.
Insulation Resistance	Over 1,000 MΩ			Between terminals and tab.
Vibration	ΔR +/- (0.25 % + 0.05 Ω)			

Fig.2.4.2.3 – Datasheet fornito dal costruttore

In Fig.2.4.2.4, è possibile vedere un particolare della resistenza presa in esame, dopo essere stata fissata alla piastra in alluminio. Si nota inoltre la presenza di un dissipatore nero alettato, i riscaldatori ed una piccola porzione della ventola utilizzata per il raffreddamento del sistema.



Fig.2.4.2.4 – Riscaldatori [14]

### 2.4.3 – Sistema di raffreddamento

Per il controllo della temperatura della piastra e dell'intero sistema, è necessario monitorare sia il livello di corrente fornito dal trasduttore di temperatura che il valore di corrente da assegnare ai riscaldatori per elevare la temperatura. Oltre a questi parametri fondamentali è importante capire quale possa essere un sistema adeguato per compiere il raffreddamento, quando la temperatura raggiunge valori indesiderati. Per fornire un adeguato controllo, è stata studiata una struttura combinata composta da un dissipatore alettato che sfrutta il principio dello scambio termico per conduzione con il sistema ambiente, ed una ventola che permette di ridurre il tempo necessario al dissipatore per raffreddarsi adeguatamente. In Fig.2.4.3 è riportato il particolare costruttivo con i componenti per il controllo della temperatura:



Fig.2.4.3 – Particolare costruttivo del controllo di temperatura

Tuttavia, la ventola deve essere in grado di accendersi e spegnersi in maniera controllata quando la temperatura supera i limiti stabiliti. Per effettuare l'operazione di accensione e spegnimento, dopo alcuni studi sul metodo da utilizzare, si è giunti alla conclusione che il dispositivo in grado di soddisfare tali richieste fosse il circuito integrato MOSFETT. Questo circuito integrato ci permette di comandare il passaggio della corrente tra drain e source, variando il funzionamento da acceso/spento con corrente erogata regolata. In questo progetto di tesi, il mosfett è utilizzato principalmente come interruttore on-off e si è scelto di utilizzare quello della casa costruttrice *INTERNATIONAL RECTIFIER IRF530N* riportato in Fig.2.4.3.1:

International  
**IR** Rectifier  
IRF530N

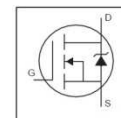


Fig.2.4.3.1 – Mosfett International Rectifier IRF530N

Secondo il datasheet del costruttore, il circuito integrato è in grado di far circolare una corrente più elevata rispetto a quella assorbita dalla ventola, imponendo una tensione di gate pari a 4.2 V. In Fig.2.4.3.2 è riportato lo schema circuitale utilizzato dal sistema che va ad alimentare il mosfett e dunque la ventola per il funzionamento desiderato. Lo schema rappresenta una piccola porzione del circuito più ampio nel quale è inserito, e del quale necessita il gruppo termo controllore per il corretto funzionamento.

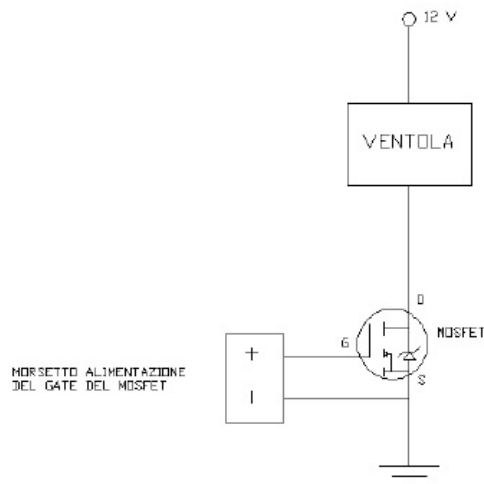


Fig.2.4.3.2 – Schema circuitale di collegamento del mosfett [14]

Con l'ausilio di un alimentatore controllato, è stato notato che il mosfett si comporta come un circuito aperto quando tra gate e source è applicata una tensione nulla, mentre si comporta come un corto circuito quando ai suoi capi è erogata una tensione di 4,2 V. Dunque fornendo una tensione nulla o una tensione pari a quella ricavata in precedenza, il mosfett assicura il controllo on-off dell'accensione della ventola. Per ovviare alla piccola seppur rilevante capacità parassita che si genera tra i terminali di gate e source, è stata introdotta una resistenza da 10 k $\Omega$  per consentire alla capacità parassita di scaricarsi. Questo permette con certezza di trovare un valore effettivamente nullo di tensione e quindi l'effettivo spegnimento della ventola.

In Fig.2.4.3.3 è riportato il mosfett utilizzato nel precedente circuito insieme alla resistenza necessaria ad annullare gli effetti parassiti tra i terminali. Questa soluzione volante, come si vedrà nei prossimi capitoli, è stata sostituita con una soluzione fissa e più affidabile tramite l'inserimento del mosfett e della resistenza nel nuovo circuito elettrico sviluppato.

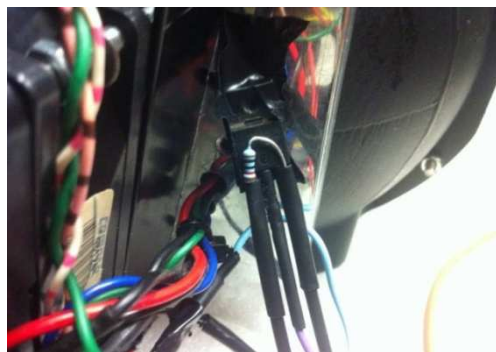


Fig.2.4.3.3 – Collegamento del mosfett e della resistenza [14]



## 2.5 – Sistema a microcontrollore Arduino

Come già accennato nei precedenti paragrafi, il sistema in esame oltre a richiedere l’ausilio di componenti elettronici per la gestione dei parametri elettrici, richiede la necessità di interagire con il terminale di comando in modo da ricevere una specifica istruzione per la lettura di un dato o l’inserimento di un setpoint. Il dispositivo scelto sia per semplicità di utilizzo e costo, sia perché risponde adeguatamente alle richieste necessarie per lo sviluppo di questo progetto, risulta essere *ARDUINO MEGA AT2560*[15]. Il dispositivo dalle ridotte dimensioni, possiede una grande versatilità di utilizzo e rende semplice la comprensione delle caratteristiche intrinseche del sistema oltre la grande quantità d’ingressi e uscite a disposizione. Sulla scheda elettronica di Arduino sono presenti morsetti di collegamento per l’utilizzo di sistemi digitali, oppure ingressi analogici e uscite in modulazione di ampiezza PWM. Attraverso il collegamento USB, è possibile interagire con Arduino andando a programmare il microcontrollore interno, in modo da far eseguire le istruzioni necessarie. Non richiede alimentazione separata per il funzionamento, anche se è presente un morsetto adatto a questo scopo. Inoltre Arduino ha al suo interno una Memoria Flash da 256 KB su cui è possibile salvare il file sorgente utilizzato dal sistema per il controllo, 4 KB di Memoria EEPROM richiamabili tramite l’utilizzo della libreria EEPROM, e 8 KB di Memoria SRAM. In Fig.2.6 è riportato un esempio di scheda a microcontrollore Arduino Mega simile a quella utilizzata in questo progetto:



Fig.2.5 – Microcontrollore Arduino Mega AT2560 [15]

## *CAPITOLO 3 - IL NUOVO CIRCUITO ELETTRICO IMPLEMENTATO- HARDWARE - SOFTWARE*

### *3.1 – Considerazioni iniziali*

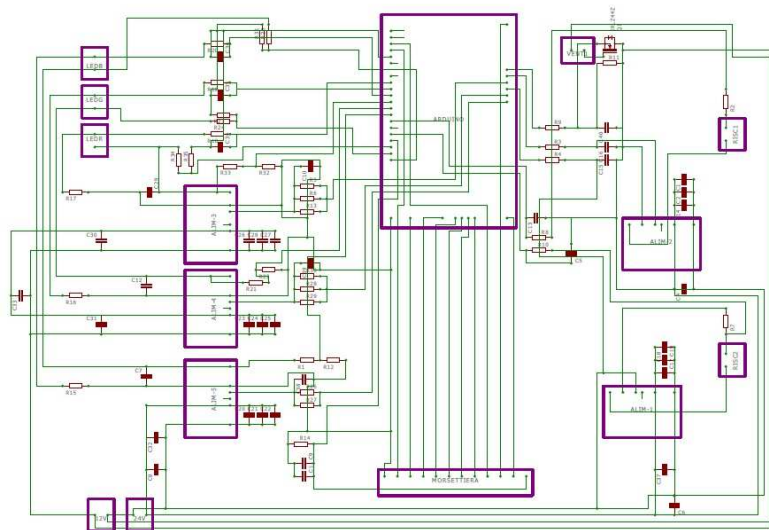
Dopo aver analizzato il sistema di partenza nel precedente capitolo, continuiamo lo sviluppo del progetto elettrico per consentire al nuovo circuito di raggiungere un livello di stabilità e affidabilità tali da rendere le misure ancora più attendibili e veritiere. Il primo step eseguito dopo l'analisi preliminare del circuito esistente, è stato quello di cercare una soluzione che potesse porre rimedio alla disposizione delle tre schede elettriche preesistenti, in modo da garantire un adeguato livello di sicurezza e affidabilità al sistema. Un'altra soluzione è stata studiata per i conduttori di collegamento dei tre elementi del circuito principale e per i conduttori che collegano il circuito alla sfera integratrice. In Fig.3.1 è riportato il vecchio sistema circuitale:



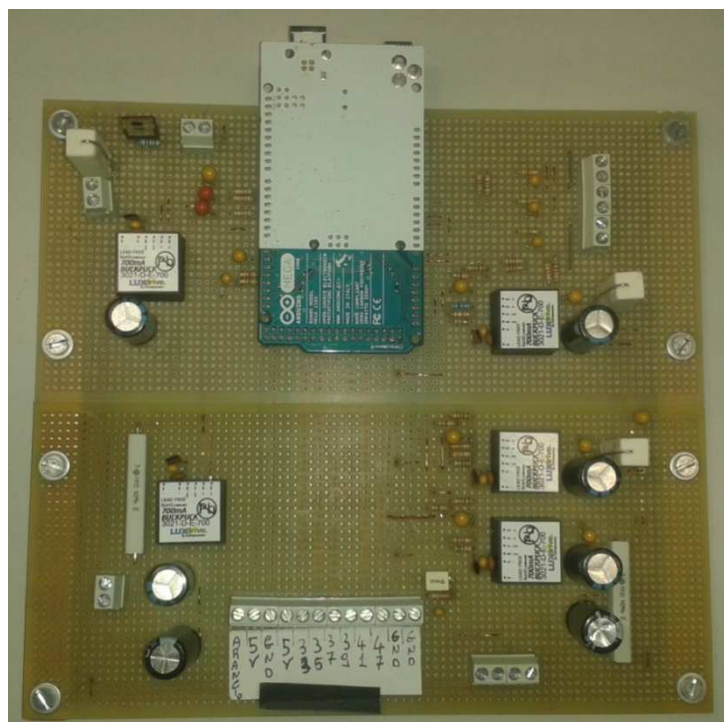
*Fig.3.1 – Circuito elettrico usato in precedenza*

### *3.2 - Hardware – Il nuovo circuito elettrico*

Il punto di partenza per questo circuito è stato quello di progettare uno schema elettrico che potesse porre rimedio alla disposizione dei componenti ed eliminare allo stesso tempo la presenza dei conduttori di collegamento tra le schede del circuito. Per rendere possibile quest'operazione, è stata studiata una soluzione a circuito unico, disponendo cioè i componenti tutti allo stesso livello sulla stessa scheda circuitale, contenente tutte le vie di collegamento. In Fig.3.2 (a) e in Fig.3.2 (b) sono riportati il nuovo schema circuitale completo e il nuovo circuito realizzato:



*Fig.3.2 (a) – Nuovo schema circuitale*



*Fig.3.2 (b) – Nuovo circuito elettrico realizzato*

A prima vista lo schema in Fig.3.2 (a) può essere confuso e poco chiaro, ma in realtà i componenti sono disposti secondo un ordine compartimentale, in modo da ottenere una suddivisione in 3 aree principali, ognuna contenente i componenti per uno specifico settore di funzionamento. In Fig.3.2.1, sono evidenziati i tre compartimenti che formano il circuito:

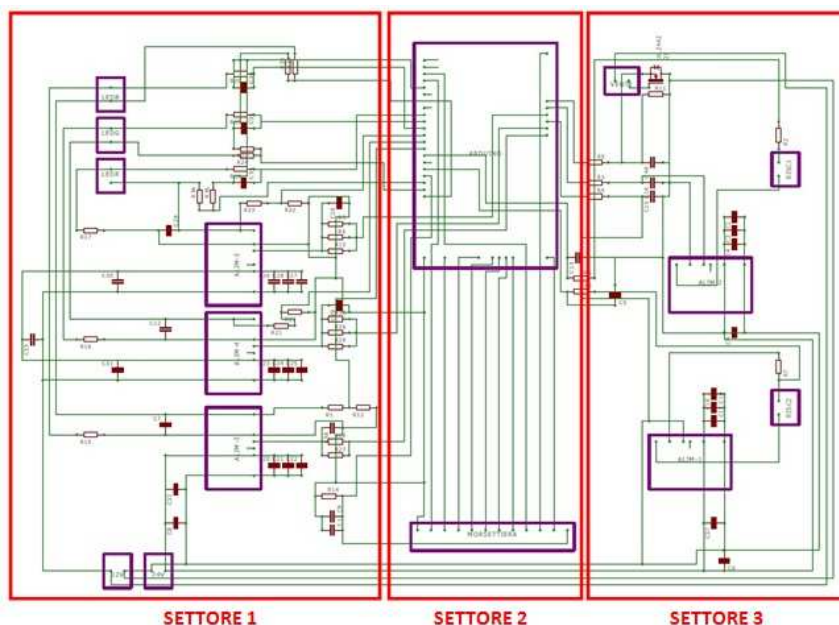


Fig.3.2.1 – Settori compartimentali del circuito

Andando ad analizzare i tre settori si possono individuare i componenti di cui essi sono composti.

### 3.3 – Settore 1 – Alimentazione e controllo dei LED

Il primo settore costituisce il circuito di alimentazione e il controllo per i parametri elettrici dei LED, in cui sono presenti anche le resistenze di filtro e i condensatori adeguati alle misure da sviluppare. Infatti come già accennato nel precedente capitolo, i segnali per le misure risultano disturbati alle alte frequenze, e dunque i filtri RC adempiono al loro scopo eliminando tali disturbi. In questo settore sono visibili i tre alimentatori DC-DC, raffigurati da rettangoli di colore viola, che erogano la corrente costante necessaria ai LED per il funzionamento. Nella parte superiore sinistra sono visibili tre morsettiere, individuate anch'esse da rettangoli di colore viola, i quali costituiscono i punti di collegamento tra il circuito e i LED, e dunque i punti di collegamento attraverso i quali scorrono la corrente necessaria ai LED per il corretto funzionamento. I condensatori elettrolitici e ceramici posti in parallelo all'ingresso degli alimentatori, costituiscono un'ottima soluzione per l'attenuazione del ripple della corrente. Inoltre in questo settore sono presenti le due morsettiere situate nella zona sottostante i collegamenti, adibite a ricevere l'alimentazione principale di 12 V e 24 V dall'alimentatore esterno regolabile per alimentare il circuito generale.

Ogni singolo componente del settore, come quelli degli altri settori, è stato calcolato e disposto in una specifica posizione, per consentire da un lato la possibilità di procedere a misure dirette sui componenti, e dall'altro facilitare il passaggio delle successive vie di collegamento tra componenti diversi. In Fig.3.3 è riportato il settore numero 1 con i componenti utilizzati e le loro grandezze:

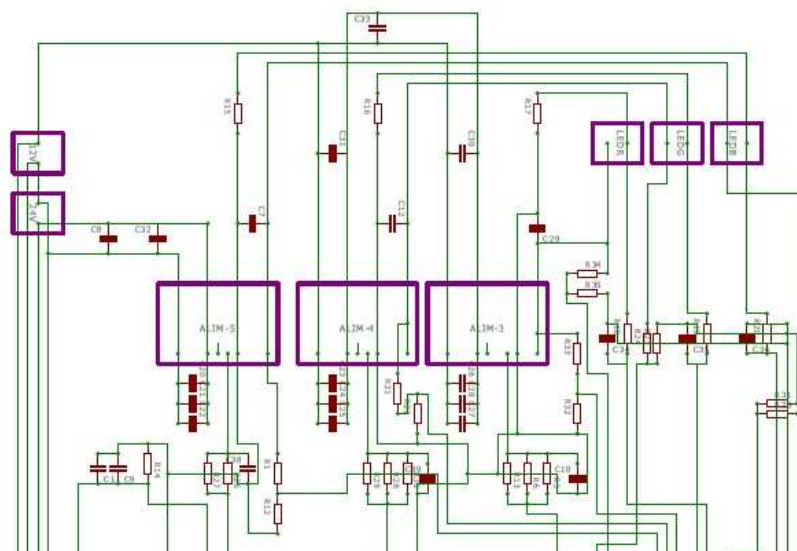


Fig.3.3 – Settore 1

I componenti presenti nel settore 1 sono:

Tabella 3.3 delle resistenze:

NOM.	DESCRIZIONE	VALORE
R1	Resistenza	270 $\Omega$ / ¼ W
R5	Resistenza	3,9 k $\Omega$ / ¼ W
R6	Resistenza	261 $\Omega$ / ¼ W
R10	Resistenza	47 k $\Omega$ / ¼ W
R12	Resistenza	4,7 K $\Omega$ / ¼ W
R13	Resistenza	150 $\Omega$ / ¼ W
R14	Resistenza	2,634 k $\Omega$ / ¼ W
R15	Resistenza ceramica	1 $\Omega$ / 7W
R16	Resistenza ceramica	1 $\Omega$ / 7W
R17	Resistenza ceramica	1 $\Omega$ / 7W
R19	Resistenza	47 k $\Omega$ / ¼ W
R20	Resistenza	47 k $\Omega$ / ¼ W
R21	Resistenza	1,5 k $\Omega$ / ¼ W
R22	Resistenza	270 $\Omega$ / ¼ W
R23	Resistenza	1,5 k $\Omega$ / ¼ W
R24	Resistenza	380 $\Omega$ / ¼ W
R25	Resistenza	1,5 k $\Omega$ / ¼ W

R26	Resistenza	270 $\Omega$ / ¼ W
R27	Resistenza	150 $\Omega$ / ¼ W
R28	Resistenza	270 $\Omega$ / ¼ W
R29	Resistenza	270 $\Omega$ / ¼ W
R30	Resistenza	3,3 k $\Omega$ / ¼ W
R31	Resistenza	380 $\Omega$ / ¼ W
R32	Resistenza	270 $\Omega$ / ¼ W
R33	Resistenza	1,5 k $\Omega$ / ¼ W
R34	Resistenza	1,5 k $\Omega$ / ¼ W
R35	Resistenza	380 $\Omega$ / ¼ W

Tabella 3.3.1 dei condensatori:

<b>NOM.</b>	<b>DESCRIZIONE</b>	<b>VALORE</b>
C1	Condensatore ceramico	22 pF
C7	Condensatore al tantalio	4,7 $\mu$ F / 35 V
C8	Condensatore elettrolitico	1000 $\mu$ F / 35 V
C9	Condensatore in poliestere	100 nF / 50 V
C10	Condensatore al tantalio	100 $\mu$ F / 3,15 V
C11	Condensatore al tantalio	103 pF
C12	Condensatore al tantalio	4,7 $\mu$ F / 35 V
C20	Condensatore ceramico	103 pF
C21	Condensatore ceramico	22 pF
C22	Condensatore ceramico	100 nF
C23	Condensatore ceramico	103 pF
C24	Condensatore ceramico	22 pF
C25	Condensatore ceramico	100 nF
C26	Condensatore ceramico	103 pF
C27	Condensatore ceramico	22 pF
C28	Condensatore ceramico	100 nF
C29	Condensatore al tantalio	4,7 $\mu$ F / 35 V
C30	Condensatore elettrolitico	220 $\mu$ F / 35 V
C31	Condensatore elettrolitico	220 $\mu$ F / 35 V
C32	Condensatore elettrolitico	220 $\mu$ F / 35 V
C33	Condensatore al tantalio	10 $\mu$ F / 35 V
C34	Condensatore al tantalio	4,7 $\mu$ F / 25 V
C35	Condensatore al tantalio	4,7 $\mu$ F / 25 V
C36	Condensatore al tantalio	4,7 $\mu$ F / 25 V
C38	Condensatore al tantalio	100 $\mu$ F / 3,15 V

I resistori e i condensatori impiegati per la formazione dei filtri RC, sono disposti in Fig.3.3 in prossimità dei collegamenti provenienti da Arduino per il controllo delle modulazioni PWM, e in prossimità dei punti essenziali utilizzati come sorgente per le misure da inviare al microcontrollore. I primi sono individuati in Fig.3.3.1 da riquadri di colore verde, gli altri da riquadri di colore celeste.

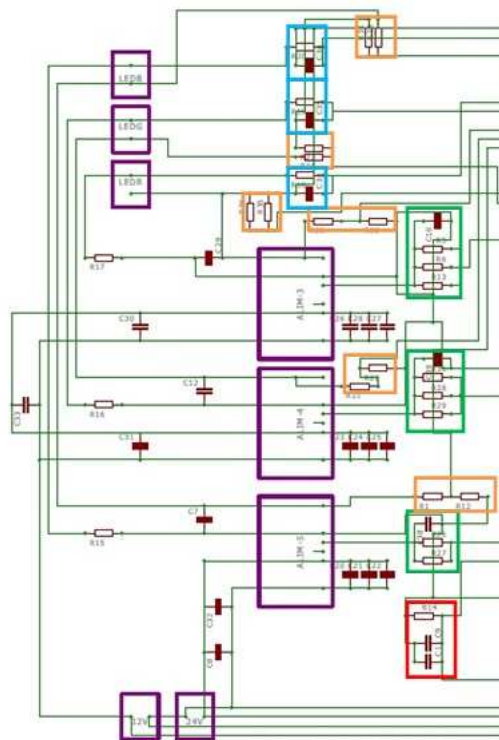


Fig.3.3.1 – Filtri e partitori per la misura

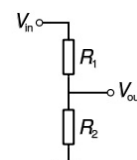
Nel settore 1 inoltre, sono presenti sei riquadri di colore marrone chiaro, che rappresentano i partitori di tensione per i segnali da mandare agli ingressi analogici di Arduino. Questi partitori sono fondamentali per fare le misure, poiché una caratteristica dei microcontrollori è quella di aumentare la precisione di calcolo quando il valore di tensione inviato agli ingressi analogici è il più vicino possibile al valore di tensione di riferimento impostato per il convertitore ADC. Dunque lo scopo di questi partitori di tensione è quello di ricevere in ingresso il valore di tensione da misurare e ridurlo nel valore nell'intorno della tensione di riferimento, prima di essere inviato al microcontrollore.

### 3.3.1 – Principio del partitore di tensione

Il partitore di tensione è costituito dalla serie di due resistenze alla quale è applicata una tensione che tenderà a ripartirsi tra di esse in funzione del valore assunto da ognuna delle due resistenze. Infatti si ha che:

la corrente che scorre in ogni resistenza è

$$I_1 = I_2 = I$$



e, dalla legge di Ohm, la tensione ai capi di ogni resistenza è

$$V_1 = I_1 * R_1 = I * R_1$$

$$V_2 = I_2 * R_2 = I * R_2$$

Sapendo che la tensione totale in ingresso  $V_{in}$  è data dalla somma delle due tensioni ai capi delle due resistenze, si ottiene:

$$I * R_1 + I * R_2 = V_{in}$$

Da questa relazione si può ricavare la corrente

$$I = \frac{V_{in}}{(R_1 + R_2)}$$

Dunque la tensione ai capi della resistenza  $R_2$  sarà pari a

$$V_{out} = V_{in} * \frac{R_2}{(R_1 + R_2)}$$

### 3.3.2 – Misura di tensione ai capi degli alimentatori dc-dc

Per procedere al calcolo delle resistenze da utilizzare nel partitore di tensione, è necessario dapprima effettuare una stima manuale del valore di tensione presente ai capi delle uscite di ogni singolo alimentatore DC-DC. Attraverso l'ausilio del multimetro presente in laboratorio, sono stati valutati i valori di tensione, nella condizione di funzionamento con la massima corrente di esercizio (SETPOINT 1000), in modo da ottenere il valore massimo di tensione. Di seguito sono riportati i risultati ottenuti:

SETPOINT 1000

$$V_{r\ mis} = 5.560 \text{ V}$$

$$V_{g\ mis} = 7.11 \text{ V}$$

$$V_{b\ mis} = 17.43 \text{ V}$$

Applicando il principio del partitore di tensione, il valore inviato all'ingresso analogico di Arduino sarà:

ALIMENTATORE LED ROSSI

$$V_{out} = V_{in} * \frac{R_2}{(R_1 + R_2)} = 5.560 * \frac{270}{(1505 + 270)} = 0.845746 \text{ mV}$$



Dunque i valori delle due resistenze sono:  $R_1 = 1.5 \text{ k}\Omega$ ,  $R_2 = 270 \Omega$

#### ALIMENTATORE LED VERDI

$$V_{out} = V_{in} * \frac{R_2}{(R_1 + R_2)} = 7.11 * \frac{270}{(1505 + 270)} = 1.08152 \text{ mV}$$

Dunque I valori delle due resistenze sono:  $R_1 = 1.5 \text{ k}\Omega$ ,  $R_2 = 270 \Omega$

#### ALIMENTATORE LED BLU

$$V_{out} = V_{in} * \frac{R_2}{(R_1 + R_2)} = 17.43 * \frac{270}{(4700 + 270)} = 0.946901 \text{ mV}$$

Dunque I valori delle due resistenze sono:  $R_1 = 4.7 \text{ k}\Omega$ ,  $R_2 = 270 \Omega$

Dopo aver calcolato il valore per ogni resistenza del partitore, questi stessi valori devono essere inseriti all'interno del firmware per permettere la decodifica inversa e ottenere il valore di tensione realmente misurato.

#### 3.3.3 – Misura della caduta di tensione ai capi della resistenza di shunt

I valori delle resistenze di shunt sono noti, mentre il valore di corrente, dipende dal setpoint impostato dall'utente. Per misurare il valore di caduta di tensione massima, è necessario anche in questo caso eseguire una misura diretta sul componente attraverso l'ausilio di un multimetro da laboratorio. Prendendo in considerazione il caso in cui è inviata la corrente massima ai LED, i valori di caduta di tensione risultano:

#### SETPOINT 1000

$$V_{r\text{mis}} = 629.6 \text{ mV}$$

$$V_{g\text{mis}} = 622.1 \text{ mV}$$

$$V_{b\text{mis}} = 527.2 \text{ mV}$$

I valori riportati, rappresentano il valore massimo di caduta di tensione che si può misurare ai capi dello shunt con corrente erogata ai LED massima.

### 3.3.4 – Calcolo della tensione presente ai capi della serie dei LED

Una delle nuove tipologie di misura introdotte dal nuovo circuito, permette di valutare la tensione presente ai capi della serie di LED, considerando la caduta di tensione ai capi della resistenza di shunt e la misura di tensione erogata in uscita dagli alimentatori DC-DC. Per compiere questa tipologia di misura, è necessario introdurre anche in questo caso, un partitore di tensione. Le resistenze che compongono il partitore, sono poste nei pressi dei punti utilizzati per effettuare le misure. I valori di tensione ai capi della serie di LED ottenuti considerando le approssimazioni introdotte dal calcolatore, risultano essere:

SETPOINT 1000

$$V_{r_{mis}} = 4.923 \text{ V}$$

$$V_{g_{mis}} = 6.468 \text{ V}$$

$$V_{b_{mis}} = 16.83 \text{ V}$$

I valori dei componenti del partitore e la tensione inviata al microcontrollore sono:

TENSIONE LED ROSSI

$$V_{out} = V_{in} * \frac{R_2}{(R_1 + R_2)} = 4.923 * \frac{270}{(1500 + 270)} = 0.750966 \text{ mV}$$

Dunque i valori delle due resistenze sono:  $R_1 = 1.5 \text{ k}\Omega$ ,  $R_2 = 270 \Omega$

TENSIONE LED VERDI

$$V_{out} = V_{in} * \frac{R_2}{(R_1 + R_2)} = 6.468 * \frac{270}{(1505 + 270)} = 0.986644 \text{ mV}$$

Dunque I valori delle due resistenze sono:  $R_1 = 1.5 \text{ k}\Omega$ ,  $R_2 = 270 \Omega$

TENSIONE LED BLU

$$V_{out} = V_{in} * \frac{R_2}{(R_1 + R_2)} = 16.83 * \frac{270}{(4700 + 270)} = 0.914306mV$$

Dunque I valori delle due resistenze sono:  $R_1 = 4.7 k\Omega$  , $R_2 = 270 \Omega$

### 3.3.5 – Calcolo della tensione misurata sul CTRL degli alimentatori dei LED

Applicando il principio appena descritto al nuovo circuito in condizioni di funzionamento con corrente massima (SETPOINT 1000) e minima (SETPOINT 0000) erogata ai LED, si ottiene:

SETPOINT 1000

SETPOINT 0000

$$V_{ctrl R} = 1.619 V$$

$$V_{ctrl R} = 4.508 V$$

$$V_{ctrl G} = 1.622 V$$

$$V_{ctrl G} = 4.508 V$$

$$V_{ctrl B} = 1.617 V$$

$$V_{ctrl B} = 4.499 V$$

Una volta misurati i valori di tensione nelle due condizioni limite, sarà utilizzato il valore massimo di tensione applicabile al Ctrl dell'alimentatore (condizione con SETPOINT 0000) per calcolare il valore delle resistenze in condizioni di funzionamento a regime.

Andando ad applicare il principio del partitore alla serie di LED rossi, verdi e blu si ottiene:

LED ROSSI

$$V_{out} = V_{in} * \frac{R_2}{(R_1 + R_2)} = 4.508 * \frac{385.4}{(1505 + 385.4)} = 0.919056 mV$$

Dunque le due resistenze del partitore risultano essere:  $R_1 = 1.505 k\Omega$  , $R_2 = 385.4 \Omega$

LED VERDI

$$V_{out} = V_{in} * \frac{R_2}{(R_1 + R_2)} = 4.508 * \frac{383.6}{(1506 + 383.6)} = 0.915151 mV$$

Dunque le due resistenze del partitore risultano essere:  $R_1 = 1.506 k\Omega$  , $R_2 = 383.6 \Omega$

LED BLU

$$V_{out} = V_{in} * \frac{R_2}{(R_1 + R_2)} = 4.499 * \frac{382.5}{(1501 + 382.5)} = 0.913654mV$$

Dunque le due resistenze del partitore risultano essere:  $R_1 = 1.501 \text{ k}\Omega, R_2 = 382.5 \Omega$

### 3.3.6 – Misura di temperatura del sistema

All'interno del settore numero uno, come si può notare in Fig.3.3, sono presenti i componenti elettronici utilizzati per la misura della temperatura del sistema, individuati dal riquadro in rosso. La temperatura viene misurata attraverso un trasduttore installato in un'apposita cavità ricavata all'interno della piastra di supporto dei LED. Questo sensore è in grado di fornire un valore di corrente proporzionale alla temperatura misurata, secondo il rapporto  $1 \mu\text{A/K}$ . Per poter utilizzare il valore di corrente ottenuto, è necessario trasformarlo in tensione da inviare ad Arduino, e ripulirlo dai disturbi introdotti dal sistema non ideale. Per questo motivo la corrente erogata dal sensore, viene fatta scorrere attraverso un'apposita resistenza, la quale genera una caduta di tensione proporzionale al valore di corrente che la attraversa. Il valore di resistenza utilizzata, dovrà poi essere inserito all'interno del firmware del microcontrollore, per poter effettuare la decodifica inversa ed ottenere il valore di temperatura reale.

Inizialmente, nel vecchio sistema è stata inserita una resistenza da  $2.2 \text{ k}\Omega$  in parallelo a due condensatori di filtro da  $103 \text{ pF}$  e  $100 \text{ nF}$ , per compiere la trasformazione e ottenere la tensione desiderata. Con lo sviluppo del nuovo progetto, questo valore di resistenza si è dimostrato essere inadeguato, poiché i valori misurati dal sensore non corrispondevano al valore della reale temperatura presente in laboratorio. Essa, infatti, è sempre stata mantenuta costante nell'intorno di  $24^\circ\text{C}$ , ma il sensore misurava valori differenti. In seguito, la resistenza da  $2.2 \text{ k}\Omega$  è stata sostituita con una nuova da  $1.156 \text{ k}\Omega$ , per riportare i valori di temperatura misurati, uguali a quelli stampati a video dal controllore, ma nonostante quest'ultima modifica, la taratura del sensore risultava ancora una volta insoddisfacente dal punto di vista numerico. Inoltre, è stata notata un'altra differenza dei valori di temperatura, quando ai LED è erogata la massima corrente. Il problema è stato ricercato all'interno del firmware di Arduino poiché si pensava fosse dovuto a un errore di scrittura del codice operativo, ma in seguito a svariate verifiche si è giunti alla conclusione che il sistema dovesse essere ulteriormente tarato.

Partendo quindi dai datasheet forniti dal costruttore del sensore, si è preso atto che il sensore alimentato con una tensione compresa tra 4 e 5 Volt, tende ad introdurre un errore di misura della corrente di circa  $0.5 \mu\text{A/K}$ . Dunque la nuova soluzione individuata per ovviare a questo problema, è stata quella di inserire una nuova resistenza da  $2.643 \text{ k}\Omega$  al posto della resistenza da  $2.156 \text{ k}\Omega$ , un ulteriore filtro composto da due condensatori da  $103 \text{ pF}$  e  $100 \text{ nF}$  nei pressi del sensore, ed inserire una costante moltiplicativa Kcalib all'interno del firmware di Arduino. In Fig.3.3.6, è riportata la piccola scheda contenente il nuovo filtro introdotto all'interno della scatola dei collegamenti in prossimità della sfera integratrice, ed il particolare del collegamento.



Fig.3.3.6 – Scheda contenente i filtri

### 3.4 – Settore 2 – Microcontrollore Arduino e collegamenti per le misure

Nel secondo settore come si può notare in Fig.3.4, non sono presenti componenti elettronici, poiché si è scelto di mantenere una zona neutra nel mezzo del circuito generale, in modo da permettere il passaggio dei conduttori di collegamento, evitando quanto più possibile la sovrapposizione delle piste di cablaggio. In Fig.3.4 si nota il circuito stampato Arduino, e le morsettiere utilizzate per ricevere le misure di temperatura e frequenze dai LED. Con riquadri di colori diversi sono stati individuati i differenti ingressi e uscite. Con il colore celeste, s’individuano gli ingressi analogici, con il riquadro verde le uscite PWM mentre con il rosso gli ingressi e le uscite digitali del sistema.

Per comprendere al meglio la tipologia di collegamenti creati, di seguito nelle Tab.3.4 (a-b-c) vengono riportati i nomi dei collegamenti ed i pin utilizzati da Arduino per gli ingressi analogici e digitali e per le uscite PWM:

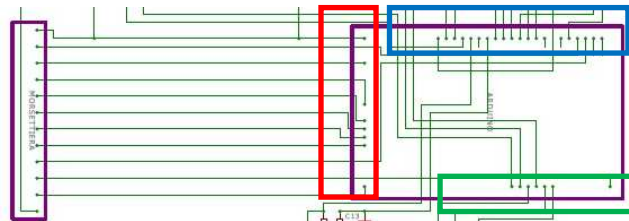


Fig.3.4 – Settore 2

Pin di Arduino che individuano gli ingressi analogici Tab.3.4 (a):

<b>INGRESSI ANALOGICI</b>	<b>DESCRIZIONE</b>
A1	Misura di corrente LED ROSSI
A2	Misura di corrente LED VERDI
A3	Misura di corrente LED BLU
A4	Misura di tensione CTRL LED ROSSI
A5	Misura di tensione CTRL LED VERDI
A6	Misura di tensione CTRL LED BLU

A7	Misura corrente al riscaldatore 1
A8	Misura corrente al riscaldatore 2
A9	Misura di temperatura
A10	Misura di tensione alimentatore LED ROSSI
A11	Misura di tensione alimentatore LED VERDI
A12	Misura di tensione alimentatore LED BLU
A13	Pin del LED interno di Arduino

Pin che individuano le uscite PWM Tab.3.4 (b):

<b>USCITE PWM</b>	<b>DESCRIZIONE</b>
PWM 2	Tensione CTRL LED ROSSI
PWM 3	Tensione CTRL LED VERDI
PWM 4	Tensione CTRL Riscaldatore 2
PWM 5	Tensione CTRL LED BLU
PWM 6	Tensione CTRL Riscaldatore 1
PWM 7	Tensione al Gate del Mosfet

Pin che individuano gli ingressi e le uscite digitali Tab.3.4 (c):

<b>PIN DIGITALI</b>	<b>DESCRIZIONE</b>
33	Pin TAOS S0
35	Pin TAOS S1
37	Pin TAOS S2
39	Pin TAOS S3
41	Pin TAOS OE
47	Pin TAOS Valore Misurato

In Fig.3.4, è presente anche la morsettiera per il collegamento dei sensori TAOS e per il sensore di temperatura. Questa morsettiera è composta di dodici punti terminali, ai quali si collegano i conduttori in arrivo dai sensori per le misure. La morsettiera è strutturata in modo tale da poter differenziare e distanziare i segnali di potenza dai segnali digitali, in modo da eliminare ogni possibile disturbo introdotto dall'intersezione dei conduttori. In Fig.3.4.1 è riportata la morsettiera con i terminali di collegamento:

Sens	5	G	5	3	3	3	3	4	4	G	G
Temp	V	N	V	3	5	7	9	1	7	N	N
		D								D	D

Fig.3.4.1 – Morsettiera di collegamento

### 3.5 – Settore 3 – Sistema di regolazione e controllo della temperatura

Il terzo ed ultimo settore di cui è composto il nuovo circuito e riportato in Fig.3.5, costituisce senz'altro un sistema essenziale per il corretto funzionamento in determinate condizioni di temperatura, per tutto il sistema. Infatti questo settore gestisce i parametri termici, attraverso l'ausilio di riscaldatori presenti sulla piastra, e di una ventola insieme al dissipatore alettato usati per asportare calore. Il principio di funzionamento utilizzato dal settore, corrisponde a quello utilizzato per alimentare e mantenere costanti i parametri elettrici dei LED. Infatti anche in questo caso, sono presenti due alimentatori DC-DC individuati dai riquadri di colore rosso, utilizzati per mantenere costante la corrente inviata ai resistori. Si nota come in questo caso i due alimentatori siano stati sovrastimati, poiché sono in grado di erogare una corrente costante fino a 700 mA, molto superiore rispetto alla massima erogata ai riscaldatori pari a 300 mA. Inoltre lo stesso principio per la misura di corrente inviata ai LED è stato utilizzato per la misura di corrente inviata ai riscaldatori, attraverso l'utilizzo dei resistori di shunt. Unica differenza circuitale presente in questo settore, è dovuta alla presenza del circuito integrato MOSFET utilizzato per comandare l'accensione e lo spegnimento della ventola. Infatti la ventola presente sulla sfera integratrice, viene comandata attraverso il controllo introdotto nel firmware di Arduino.

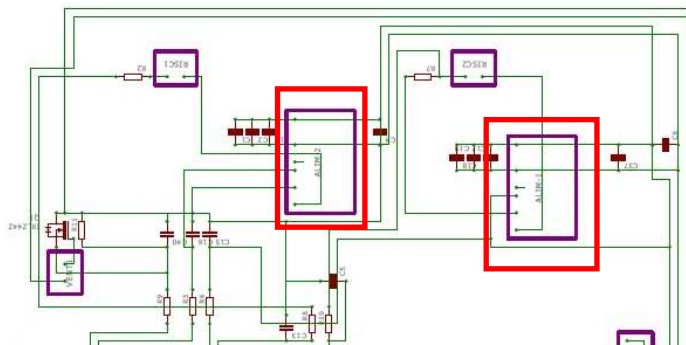


Fig.3.5 – Settore 3

Anche in questo caso sono presenti condensatori connessi in parallelo agli ingressi degli alimentatori, in modo da ridurre le fluttuazioni di corrente e rendere costante il più possibile il valore erogato. Nelle Tab.3.5 (a-b) sono riportati i componenti utilizzati in questo settore:

Tabella delle resistenze Tab.3.5 (a):

<b>NOM.</b>	<b>DESCRIZIONE</b>	<b>VALORE</b>
R2	Resistenza ceramica	1 $\Omega$ / 7W
R3	Resistenza	270 $\Omega$ / ¼ W
R4	Resistenza	270 $\Omega$ / ¼ W
R7	Resistenza ceramica	1 $\Omega$ / 7W
R8	Resistenza	47 k $\Omega$ / ¼ W
R9	Resistenza	270 $\Omega$ / ¼ W
R10	Resistenza	47 k $\Omega$ / ¼ W
R11	Resistenza	10 k $\Omega$ / ¼ W

Tabella dei condensatori Tab. 3.5 (b):

<b>NOM.</b>	<b>DESCRIZIONE</b>	<b>VALORE</b>
C1	Condensatore ceramico	22 pF
C2	Condensatore ceramico	100 nF
C4	Condensatore elettrolitico	220 $\mu$ F / 35 V
C5	Condensatore al tantalio	4,7 $\mu$ F / 25 V
C6	Condensatore elettrolitico	1000 $\mu$ F / 50 V
C13	Condensatore al tantalio	4,7 $\mu$ F / 35 V
C14	Condensatore ceramico	103 pF
C15	Condensatore al tantalio	100 $\mu$ F / 3,15 V
C16	Condensatore al tantalio	100 $\mu$ F / 3,15 V
C17	Condensatore ceramico	103 pF
C18	Condensatore ceramico	22 pF
C19	Condensatore ceramico	100 nF
C37	Condensatore elettrolitico	220 $\mu$ F / 35 V
C40	Condensatore al tantalio	100 $\mu$ F / 6,30 V

### 3.6 – Cavo di collegamento tra il nuovo circuito e il sistema contenente i LED

Una tra le principali cause che provocano un disturbo entrante nei segnali per le misure, è senza dubbio attribuibile al cavo di collegamento utilizzato per connettere fisicamente il circuito elettrico, con il sistema contenente i LED e i sensori per le misure. Il vecchio cavo infatti, non permetteva una facile gestione e movimentazione del sistema, poiché era costituito da un fascio di conduttori, di diverso colore e sezione, molto spesso collegati tra loro tramite molteplici giunzioni, quindi fonte di disturbo e malfunzionamenti. Inoltre erano presenti anche collegamenti inadeguati all'ingresso della sfera integratrice, il quale causava disordine e incomprendimento durante la fase di misura diretta sui componenti. Per ovviare a tale situazione scomoda, è stata studiata una soluzione vantaggiosa, che permette da un lato di sostituire il cavo principale con uno avente conduttori tutti alla stessa sezione e lunghezza, e dall'altro introduce la possibilità di creare due sistemi perfettamente indipendenti, attraverso l'ausilio di una porta di collegamento seriale. In questo modo tramite la porta



contenente più pin a disposizione, è possibile in una sola operazione, disconnettere e connettere la sfera integratrice dal circuito elettrico, per permettere ulteriori modifiche e misure senza rischiare di danneggiare la componentistica. Nelle Fig.3.6 (a-b) sono riportati i collegamenti prima e dopo l'introduzione del nuovo cavo, con il particolare della nuova porta utilizzata:



*Fig.3.6 (a) Conduttori precedenti – (b) Nuovi collegamenti*

In Fig.3.6.1 è riportato il particolare della spina di collegamento utilizzata. Come si può notare, questa spina offre la possibilità di connettere più di un conduttore contemporaneamente, permettendo il perfetto isolamento dei due componenti in questione, facilitando le procedure di misura e movimentazione.



*Fig.3.6.1 – Spina di collegamento*

In Tab.3.6 sono riportati i corrispondenti pin di collegamento del cavo e della sfera, che permettono di individuare il tipo di conduttore e la sua funzione.

<b>Tabella esplicativa dei pin delle morsettiere e dei cavi di collegamento</b>		
<b>PIN</b>	<b>Numerazione morsettiere lato sfera</b>	<b>Numerazione morsettiere lato cavo di collegamento</b>
<b>1</b>	Conduttore viola di <b>Rs2 (+)</b>	Conduttore <b>rosso</b> del cavo
<b>2</b>	Conduttore verde dei <b>led verdi (+)</b>	Conduttore <b>verde scuro</b> del cavo
<b>3</b>	Conduttore blu dei <b>led blu (+)</b>	Conduttore <b>verde</b> del cavo
<b>4</b>	Conduttore marrone di <b>Rs1 (-)</b>	Conduttore <b>grigio</b> del cavo
<b>5</b>	Conduttore rosso della <b>ventola (+)</b>	Conduttore <b>bianco</b> del cavo
<b>6</b>	Conduttore nero della <b>ventola (-)</b>	Conduttore <b>viola</b> del cavo
<b>7</b>	Conduttore verde di <b>Rs1 (+)</b>	Conduttore <b>arancio</b> del cavo
<b>8</b>	Conduttore nero <b>GND</b>	Conduttore <b>marrone scuro</b> del cavo
<b>14</b>	Conduttore giallo di <b>Rs2 (-)</b>	Conduttore <b>rosso/blu</b> del cavo
<b>15</b>	Conduttore nero dei <b>led verdi (-)</b>	Conduttore <b>blu</b> del cavo
<b>16</b>	Conduttore rosso dei <b>led rossi (+)</b>	Conduttore <b>giallo</b> del cavo
<b>17</b>	Conduttore nero dei <b>led rossi (-)</b>	Conduttore <b>rosa</b> del cavo
<b>18</b>	Conduttore nero dei <b>led blu (-)</b>	Conduttore <b>nero</b> del cavo
<b>19</b>	Conduttore del <b>sensore di temperatura</b>	Conduttore <b>celeste</b> del cavo da collegare al <b>pin in uscita denominato arancione</b>
<b>20</b>	Conduttore di <b>alimentazione del sensore di temperatura (+5V)</b>	Conduttore <b>marrone</b> del cavo da collegare al <b>pin di alimentazione +5V</b>

Tab.3.6 – Collegamenti del cavo e delle spine



### 3.7 – Software – Modifiche apportate al firmware di Arduino

Dopo aver analizzato in dettaglio le sostanziali modifiche apportate all'hardware preso in esame durante questo lavoro di tesi, per far sì che le nuove tipologie di misure introdotte potessero essere valutate correttamente dall'utente, si è resa necessaria l'implementazione sia del precaricato firmware di Arduino, che la scrittura di una nuova porzione di codice. Infatti come già accennato, alcuni miglioramenti al circuito, richiedono nuove misure dei componenti inseriti e il corretto interfacciamento tra questi componenti e il nuovo codice per

il controllore. Alcune modifiche, hanno richiesto la ricompilazione del codice presente in diverse funzioni collegate tra loro, e la modifica sostanziale di una parte di queste. Nel seguito vengono esaminate in dettaglio le modifiche effettuate, e il codice utilizzato per le nuove misure inserite.

### 3.7.1 – Misura di tensione al pin CTRL dell'alimentatore dc-dc

Per la misura della reale tensione di controllo  $V_{ctrl}$  presente ai capi dell'alimentatore dei LED, viene implementato il circuito collegando i rispettivi pin agli ingressi analogici. In realtà, questo valore di tensione, viene calcolato attraverso la caratteristica di controllo ricavata per gli alimentatori ed impostata nel firmware, come accadeva nella versione precedente l'implementazione. Il procedimento è reso necessario poiché il valore di tensione calcolato, è utilizzato all'interno della funzione *ControlloPid()* per il controllo della corrente inviata ai LED.

Il controllo delle correnti, eseguito ciclicamente tramite il ciclo *Voidloop()* nel firmware, esegue una differenza tra il setpoint di corrente impostato dall'utente tramite terminale, e la corrente che viene erogata ai LED attraverso la misura di tensione alla resistenza di shunt. Se la differenza tra questi due valori è nulla, si avranno correnti uguali, altrimenti un valore diverso da zero rappresenta l'errore proporzionale del ciclo. In seguito attraverso la somma dell'errore proporzionale del ciclo in esame e quello del ciclo precedente, viene individuato l'errore integrale. Per calcolare l'errore derivato invece, viene misurata la velocità di variazione della corrente tra il ciclo in esame e quello precedente. Una volta individuati gli errori e pesati con i propri coefficienti moltiplicativi, il loro valore complessivo sarà sommato al valore di corrente impostato dall'utente. In questo modo la corrente impostata e quella misurata sono uguali, poiché viene sommato di volta in volta il valore di corrente ricavato come errore, di cui il setpoint si discosta dal valore atteso.

Per compensare questa corrente in maniera ciclica, all'interno del firmware di Arduino, è stata impostata una caratteristica per gli alimentatori che calcola di quanto la tensione ai capi del pin CTRL deve aumentare, per ottenere il valore di corrente ricavato dall'errore.

### 3.7.1.2 – Modifiche apportate e problemi riscontrati

Durante le misure effettuate, i valori dei parametri misurati, risultano essere diversi dai valori attesi precedentemente calcolati. Infatti, impostando un setpoint di corrente nullo al terminale di controllo, i LED non rispondevano adeguatamente. Le cause di tale discrepanza sono state individuate sia sul lato software che sul lato hardware. Per individuare la soluzione da apportare, è stato acquistato un nuovo circuito Arduino, in modo da poter avere una scelta intercambiabile tra due sistemi diversi ed effettuare ulteriori confronti delle misure.

La prima causa di errore è stata individuata nella differente tensione di riferimento interna per il convertitore ADC a 10 bit del nuovo Arduino, diversa da 1.1 V, impostata nel setup iniziale del vecchio circuito. Alla tensione di riferimento 1.1V corrisponde la massima risoluzione del convertitore pari a  $10\text{ bit} = 2^{10} - 1 = 1023$ , risoluzione che viene utilizzata da

Arduino per convertire i valori di tensione e corrente in un numero proporzionale alla tensione di riferimento, secondo la proporzione:

$$\frac{1023}{1.1} * Valore = N$$

Il nuovo Arduino invece, presenta una tensione di riferimento pari a 1.103 V, diversa dal precedente valore di una quantità da poter influire sulle misure.

La seconda causa di errore è stata individuata nei parametri impostati all'interno della funzione che lega la corrente alla tensione del pin CTRL. Attraverso il firmware viene imposto al componente, un funzionamento nel tratto lineare della caratteristica che permette di ottenere la corrente massima di 0.7 A, applicando una tensione al pin CTRL pari a 1.65 V, mentre per ottenere una corrente erogata nulla, la tensione da applicare è di 4.2 V. Si può dunque esprimere la funzione che lega corrente erogata e tensione da applicare Fig.3.7.1.2 con:

$$V_{ctrl} = -3.6 * I_{out} + 4.2$$

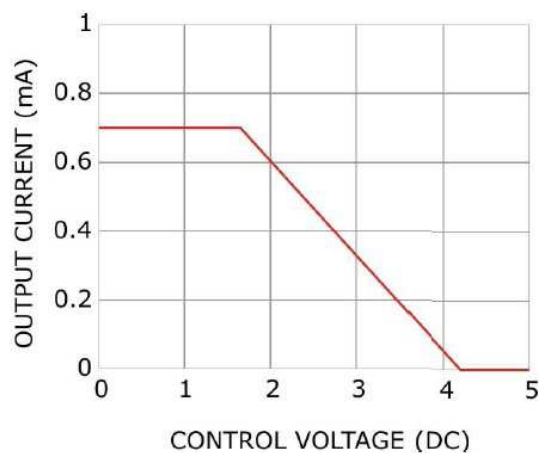


Fig.3.7.1.2 – Caratteristica Corrente - Tensione

Questa funzione dopo la modifica della tensione di riferimento per il convertitore ADC, risulta essere inadeguata, poiché i valori estremi del range di funzionamento, non permettono più l'erogazione nulla e massima della corrente. Compiendo misure dirette sul componente con setpoint di corrente massimo e setpoint di corrente nulla, sono stati individuati i valori del nuovo range di funzionamento pari a:

$$V_{min} = 1.735 V e V_{max} = 4.503 V$$

Ottenuti questi valori, è stata calcolata la nuova funzione che lega la corrente erogata alla tensione  $V_{ctrl}$  riportata di seguito:

$$V_{ctrl} = -3.755 * I_{out} + 4.248$$

La terza ed ultima causa di errore individuata nel codice, risulta essere una conseguenza delle precedenti. Infatti, come già espresso, la tensione  $V_{ctrl}$  fornita all'alimentatore attraverso la funzione *ImpostaPWM()*, è erogata dall'uscita PWM di Arduino, in funzione di una conversione effettuata tramite il convertitore ADC, ed in funzione della correzione dell'errore di corrente individuato dalla funzione *ControlloPID()*. Essendo cambiati i valori di tensione di riferimento e i valori estremi della caratteristica dell'alimentatore, risulta cambiato anche il valore del numero corrispondente alla risoluzione dell'ADC secondo la relazione:

$$\frac{1023}{4.248} * V_{ctrl} = N$$

Di seguito è riportato il nuovo codice implementato nel firmware:

```
float mDriverLed = -3.755238142;
```

```
float qDriverLed = 4.248;
```

\*\*\*\*\* **FUNZIONE CHE IMPLEMENTA UN CONTROLLORE PID** \*\*\*\*\*

```
void ControlloPid(intcolore)
{
  if (ternaRGB[colore] == 0 )
  {
    correnteMisurataVecchia[colore] = correnteMisurata[colore];
    MisuraCorrente(colore);
    errore[colore] = ternaRGB[colore] - correnteMisurata[colore];
    erroreIntegrale[colore] += errore[colore];
    erroreDerivato[colore] = (correnteMisurata[colore] - correnteMisurataVecchia[colore]) /
    dt;
    if (primociclo == 1)
    {
      erroreDerivato[colore] = 0;
      primociclo = 0;
    }
    correzione[colore] = 0;
    tensioneCtrlCalcolata[colore] = qDriverLed;
  }
  else
  {
    correnteMisurataVecchia[colore] = correnteMisurata[colore];
```

```

MisuraCorrente(colore);
    errore[colore] = ternaRGB[colore] - correnteMisurata[colore];
erroreIntegrale[colore] += errore[colore];
erroreDerivato[colore] = (correnteMisurata[colore] - correnteMisurataVecchia[colore]) /
dt;
if (primociclo == 1)
    {
    erroreDerivato[colore] = 0;
    primociclo = 0;
    }
    correzione[colore] = errore[colore] * kp + erroreIntegrale[colore] * ki +
erroreDerivato[colore] * kd;
    correzione[colore] += ternaRGB[colore];
tensioneCtrlCalcolata[colore] = mDriverLed * correzione[colore] + qDriverLed;
    }
if (tensioneCtrlCalcolata[colore] <= 1.619333)
    {
    tensioneCtrlCalcolata[colore] = 1.619333;
    }
if (tensioneCtrlCalcolata[colore] >= 4.248)
    {
    tensioneCtrlCalcolata[colore] = 4.248;
    }
CalcolaPWM(colore);
ImpostaPWM(colore);
}

```

### ***FUNZIONE CHE IMPOSTA LA PWM IN FUNZIONE DEL COLORE SELEZIONATO***

```

voidImpostaPWM(int colore)
{
    // Variabili locali
    intpinPWM = 0;
    if (colore == rosso)
    {
    pinPWM = pinRosso;
    }
    if (colore == verde)
    {
    pinPWM = pinVerde;
    }
    if (colore == blu)
    {
    pinPWM = pinBlu;
    }
}

```

```

    Timer3.pwm(pinPWM, ternaPWM[colore]);
}

```

**\*\* FUNZIONE CHE CALCOLA IL VALORE DI PWM DA IMPOSTARE AI LED \*\***

```

void CalcolaPWM(intcolore)
{
    float kLed = 240.8192090;
    ternaPWM[colore] = tensioneCtrlCalcolata[colore] * kLed;
}

```

Come già accennato nei paragrafi precedenti, per eseguire la misura di tensione presente ai capi della serie di LED, è necessario conoscere sia la caduta di tensione presente ai capi della resistenza di shunt, che la tensione presente ai terminali dell'alimentatore ai quali saranno poi collegate le serie di LED.

### 3.7.2 – Misura di tensione erogata dagli alimentatori dei LED

Per misurare questa tensione, è stato necessario introdurre all'interno del nuovo circuito, un partitore di tensione che permette di ridurre il valore di tensione misurato ai capi dell'uscita dell'alimentatore, fino ad un valore ideale per l'elaborazione che andrà poi ad effettuare Arduino. Dunque all'interno della porzione di firmware sviluppato per questa misura, si dovrà tener conto dei valori di resistenze utilizzate dal partitore, per ogni alimentatore. Inoltre si dovrà specificare nel firmware, quali pin sono utilizzati come ingressi analogici, per permettere alla libreria *analogRead()*, di andare a leggere i valori desiderati. Gli ingressi adibiti alla misura di questi valori sono:

A10	Misura di tensione alimentatore LED ROSSI
A11	Misura di tensione alimentatore LED VERDI
A12	Misura di tensione alimentatore LED BLU

Inoltre bisogna inserire all'interno del firmware il valore delle due resistenze utilizzate dal partitore per ogni alimentatore. Di seguito sono riportati i valori delle resistenze corrispondenti espressi in Ohm:

$r_{PonteR1} = 1448 \Omega$ ; //resistenza R1 del partitore di tensione del colore Rosso sull'alimentatore

$r_{PonteR2} = 250.9 \Omega$ ; //resistenza R2 del partitore di tensione del colore Rosso sull'alimentatore

```

gPonteR1 = 1448 Ω; // " R1 " " " Verde sull'alimentatore
gPonteR2 = 251.3 Ω; // " R2 " " " Verde sull'alimentatore
bPonteR1 = 4498 Ω; // " R1 " " " Blu sull'alimentatore
bPonteR2 = 258.3 Ω; // " R2 " " " Blu sull'alimentatore

```

Essendo il valore di tensione in ingresso ad Arduino, inferiore rispetto al valore misurato attraverso il multimetro per via della presenza del partitore di tensione, all'interno della funzione *MisuraTensione()* destinata ad eseguire questa misura, viene fornito come costante moltiplicativa, il valore totale del partitore già invertito, in modo da facilitare il calcolo ad Arduino.

Il valore della tensione inviata ad Arduino è dato dalla relazione:

$$V_{out} = V_{in} * \frac{R_2}{(R_1 + R_2)}$$

Dunque per ottenere la lettura del valore di tensione sul terminale degli alimentatori, la relazione riportata sopra diventa:

$$V_{in} = V_{out} * \frac{(R_1 + R_2)}{R_2}$$

Quindi I valori espressi in Ohm del partitore, forniti come costanti moltiplicative diventano:

```

rPartTot = 6.77122 Ω; //valore totale della resistenza del partitore di tensione
dell'alimentatore rosso

```

```

gPartTot = 6.76204 Ω; //valore totale della resistenza del partitore di tensione
dell'alimentatore verde

```

```

bPartTot = 18.4139 Ω; //valore totale della resistenza del partitore di tensione
dell'alimentatore blu

```

Di seguito è riportato il codice che esegue la misura di tensione erogata dagli alimentatore dei LED:

```

***** FUNZIONE CHE MISURA LA TENSIONE EROGATA DAGLI
ALIMENTATORI DEI LED*****

```

```

void MisuraTensione(int colore)

```

```

{

```

```

int lettura = 100;

```



```

float risoluzione = 0.0009775171;
int pin = 0;
tensioneAlimentatoriMisurata[colore] = 0;
if (colore == rosso)
{
    pin = tensioneRosso;
}
if (colore == verde)
{
    pin = tensioneVerde;
}
if (colore == blu)
{
    pin = tensioneBlu;
}
for (int i = 0; i < letture; i++)
{
    tensioneAlimentatoriMisurata[colore] += analogRead(pin);
}
tensioneAlimentatoriMisurata[colore] /= letture;
tensioneAlimentatoriMisurata[colore] *= risoluzione * VREF * partitore[colore];
}

```

### 3.7.3 – Misura della caduta di tensione ai capi della resistenza di shunt

Per effettuare la misura della caduta di tensione ai capi della resistenza di shunt dei LED, è necessario conoscere sia il valore della resistenza di shunt, che il valore di corrente che attraversa la resistenza stessa. La funzione che misura la corrente erogata ai LED, è individuata nel firmware con il nome *MisuraCorrente()*. Per effettuare questa misura, è stato sufficiente creare una funzione che tenga in considerazione questi due parametri per effettuare una moltiplicazione. La misura di corrente erogata ai LED varia in funzione del setpoint impostato dall'utente al terminale *TeraTerm*, mentre il valore delle resistenze è stato misurato attraverso il multimetro presente in laboratorio.

Di seguito sono riportati i valori delle resistenze di shunt espressi in Ohm, e i codici per la misura della corrente e per la misura della caduta di tensione.

```

rRosso = 1.003 Ω;
rVerde = 1.013 Ω;
rBlu = 1.019 Ω;

```

**\*\*\*\*\* FUNZIONE CHE MISURA LA CORRENTE EROGATA AI LED \*\*\*\*\***

```
voidMisuraCorrente(int colore)
{
  int letture = 100;
  float risoluzione = 0.0009775171;
  int pin = 0;
  correnteMisurata[colore] = 0;
  if (colore == rosso)
  {
    pin = correnteRosso;
  }
  if (colore == verde)
  {
    pin = correnteVerde;
  }
  if (colore == blu)
  {
    pin = correnteBlu;
  }
  for (int i = 0; i < letture; i++) {
    correnteMisurata[colore] += analogRead(pin);
  }
  correnteMisurata[colore] /= letture;
  correnteMisurata[colore] *= risoluzione * VREF;
  correnteMisurata[colore] /= resistenza[colore];
}
```

**\*\*\*\*\* FUNZIONE CHE CALCOLA LA CADUTA DI TENSIONE AI CAPI DELLO SHUNT DEI LED \*\*\*\*\***

```
voidMisuraTensioneShunt(int colore)
{
  tensioneShuntCalcolata[colore] = 0; //Variabile globale dove andiamo ad inserire il valore
  di tensione degli shunt
  MisuraCorrente(colore);
  tensioneShuntCalcolata[colore] = resistenza[colore] * correnteMisurata[colore];
}
```

#### 3.7.4 – Misura di tensione ai capi della serie di LED

Una volta ottenuti i valori di tensione erogata dagli alimentatori e il valore della caduta di tensione in funzione della corrente erogata, il calcolo di tensione presente ai capi della serie di LED, risulta essere una semplice sottrazione tra questi due valori. Dunque la funzione *MisuraTensioneLed()* implementata nel firmware di Arduino, risulta molto semplice ed

intuitiva. Inoltre trovano riscontro i valori ottenuti al terminale, andando ad effettuare le misure dirette sui componenti del circuito.

Di seguito in Tab.3.7.4 sono riportati i valori di tensione descritti in precedenza, ottenuti tramite misure dirette con multimetro sui componenti, con un setpoint di corrente massima impostato al terminale:

SETTPOINT 1000

<b>Erogata dagli alimentatori [V]</b>	<b>Pin di collegamento [V]</b>	<b>Caduta di tensione sullo shunt [mV]</b>
$V_r = 5.560 V$	$V_r = 4.923 V$	$V_r = 637.6 mV$
$V_g = 7.11 V$	$V_g = 6.468 V$	$V_g = 642.1 mV$
$V_b = 17.43 V$	$V_b = 16.83 V$	$V_b = 601.2 mV$

Tab.3.7.4 – Valori di tensione misurata con multimetro

Di seguito è presente il codice implementato nel firmware che esegue il calcolo della tensione ai capi dei LED:

\*\*\*\*\* **FUNZIONE CHE CALCOLA LA TENSIONE SUI LED** \*\*\*\*\*

```
voidMisuraTensioneLed(int colore)
{
  tensioneLedCalcolata[colore] = 0;
  MisuraTensioneShunt(colore);
  MisuraTensione(colore);
  tensioneLedCalcolata[colore] = tensioneAlimentatoriMisurata[colore] -
  tensioneShuntCalcolata[colore];
}
```

### 3.7.5 – Introduzione del setpoint di temperatura

Come ormai noto dai molteplici studi eseguiti sulle diverse cause che provocano un degrado prestazionale nell'emissione dei LED, l'elevato calore sviluppato dalla giunzione è senz'altro la causa che provoca circa l'80% di dissipazione dell'energia fornita al LED, facendo aumentare la temperatura di giunzione e quindi l'aumento del degrado prestazionale.

Non potendo avviare a tale situazione, si è scelto di implementare all'interno del sistema creato, un controllo di temperatura, come già accennato nei paragrafi precedenti.

Attraverso un controllo è possibile mantenere la temperatura del sistema, all'interno di un range di valori oltre i quali si attivano i dispositivi adibiti a riscaldare o raffreddare il sistema. Infatti sia nel precedente circuito, che nel nuovo, è stato mantenuto il controllo sviluppato e basato su due concetti molto semplici:

- Riscaldamento il sistema: se la temperatura raggiunge un valore più basso del minimo impostato;
- Raffreddamento il sistema: se la temperatura supera un valore massimo impostato;

È stato scelto come valore ottimale di funzionamento, la temperatura di 29°C, poiché è il valore raggiunto dall'intero sistema a regime, quando ai LED è impostata la massima corrente di esercizio. Lo schema riportato in Fig.3.7.5, esprime il funzionamento del controllo prima della sua implementazione:

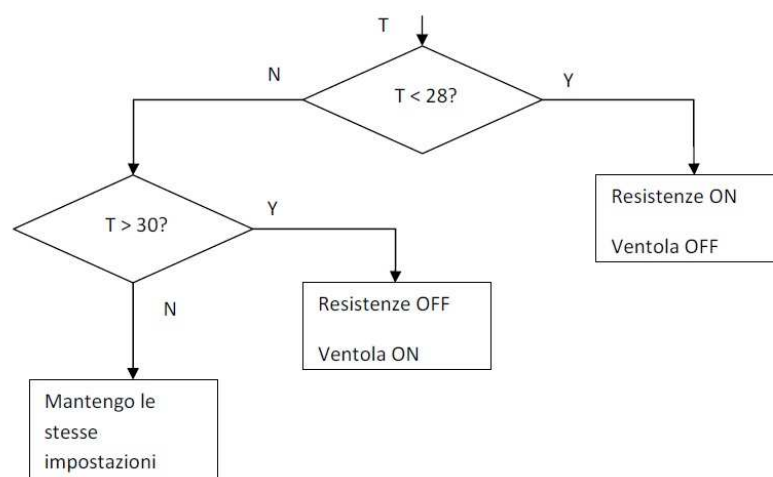


Fig.3.7.5 – Flow Chart del controllo di temperatura[14]

Lo scopo dell'introduzione del setpoint di temperatura, è quello di permettere al sistema di lavorare nell'intorno di un valore di temperatura scelto dall'utente, in modo da effettuare misure e valutare le prestazioni in determinati range. Infatti il controllo preimpostato e descritto nel flow chart precedente, è rimasto pressoché uguale, con l'aggiunta di un miglioramento dovuto a questa nuova impostazione inserita.

Il primo step eseguito per poter introdurre il setpoint descritto, è stato quello di andare a tarare perfettamente il trasduttore di temperatura utilizzato, con il nuovo firmware e il nuovo valore di resistenza inserita nel circuito. Nei paragrafi precedenti è già stato descritto come il trasduttore utilizzato, eroga ad una resistenza presente sul circuito, un valore di corrente

espresso in microampere proporzionale al valore di temperatura misurato, convertendolo poi in un valore di tensione da inviare ad Arduino.

Di seguito viene riportata la funzione *MisuraTemperatura()* adibita alla misura richiesta, in cui viene inserito il nuovo valore di resistenza utilizzato, e una costante *Kcalib = 0.9898* necessaria per effettuare la calibrazione assoluta della misura in termini di guadagno:

**\*\*\*\*\* FUNZIONE CHE MISURA LA TEMPERATURA DELLA PIASTRA \*\*\*\*\***

```
void MisuraTemperatura()
{
  resistenzaTemp = 2.643;
  float risoluzione = 0.0009775171;
  int letture = 1000;
  float kcalib = 0.9898; //coefficiente di calibrazione assoluta della misura
  for (int i = 0; i < letture; i++)
  {
    temperatura += analogRead(pintemperatura);
  }
  temperatura /= letture;
  numero = temperatura;
  temperatura *= risoluzione * VREF;
  temperatura /= resistenzaTemp;
  temperatura *= 1000. *kcalib;
  temperatura -= 273.15;
```

Una volta migliorata la calibrazione dei componenti, la misura effettiva di temperatura del sistema, risulta essere coerente con i valori misurati dai termometri nei multimetri digitali e da quelli presenti in laboratorio, con un range di errore massimo nell'intorno della misura di  $\pm 0.5 \mu\text{A/K}$ . Questo valore massimo di scostamento dalla misura reale, è perfettamente in linea con le disposizioni descritte dai datasheet del sensore, fornito dal costruttore.

Il secondo step eseguito è stato quello di andare a introdurre il comando nel firmware, che permette di impostare un valore di temperatura desiderato. Il principio utilizzato per quest'operazione, risulta essere simile a quello introdotto dal setpoint di corrente ai LED. Infatti il setpoint di temperatura, introduce un valore di corrente da fornire ai riscaldatori per raggiungere il valore richiesto, in sincronismo con il controllo di temperatura sviluppato. Al sistema, non è possibile richiedere il funzionamento nell'intorno di un valore di temperatura inferiore a quella presente nell'ambiente, per questo motivo all'interno del firmware è inserito un valore di temperatura al quale il sistema si porterà automaticamente tramite il controllo imposto. Il valore di temperatura di setpoint fisso interno è stato scelto intorno ai 25°C come la temperatura presente in laboratorio, per permettere l'inizio del funzionamento senza dover prima riscaldare o raffreddare il sistema. Verrà in seguito valutato il valore del setpoint inserito e saranno eseguite due richieste dal firmware come quelle espresse in precedenza e riportate di seguito:

- temperatura  $< \text{valoreTempImp} - \text{semiAmpiezzaIsteresi}$ : accendo i riscaldatori, spengo la ventola
- temperatura  $\geq \text{valoreTempImp} + \text{semiAmpiezzaIsteresi}$ : spengo i riscaldatori, accendo la ventola

I parametri riportati valgono:

temperatura = 20°C, valore fisso interno al firmware;  
 valoreTempImp = [ 0 – 99 ] °C, range di valori scelti dall'utente tramite il setpoint;  
 semiAmpiezzaIsteresi =  $\pm 0.25$  °C, rappresenta un range di errore ammissibile per il setpoint inserito;

Di seguito viene riportato il codice sviluppato ed inserito nel firmware, per la lettura dei valori di setpoint di temperatura:

**\*\* CODICE PER LA LETTURA DEL SETPOINT DI TEMPERATURA RICHIESTO \*\***

```

if (leggiStringa[2] == 116)
  {
  int k = 1;
  int calcolato = 0;
  intsecondoDato = 0;
  if (leggiStringa[2 + (k + 1)] == CR && calcolato == 0)
    {
    int provvisorio1 = leggiStringa[2 + k] - 48;
    leggi = provvisorio1;
    secondoDato = 6;
    calcolato = 1;
    valoreTempImp = map (leggi, 0, 99, 0, 99);
    }
  if (leggiStringa[2 + (k + 2)] == CR && calcolato == 0)
    {
    int provvisorio1 = leggiStringa[2 + k] - 48;
    int provvisorio2 = leggiStringa[2 + (k + 1)] - 48;
    leggi = provvisorio1 * 10 + provvisorio2;
    secondoDato = 7;
    calcolato = 1;
    valoreTempImp = map (leggi, 0, 99, 0, 99);
    }
  }

```

Terzo ed ultimo step eseguito per chiudere il processo di misura, impostazione e controllo della temperatura del sistema, è stato quello di andare a modificare il ciclo if eseguito all'interno della funzione *MisuraTemperatura()*, introducendo i parametri espressi

precedentemente, per l'esecuzione del controllo. Questo ciclo è inserito all'interno di *voidLoop()* nel firmware di Arduino, per permettere di eseguire automaticamente all'accensione il controllo impostato. Infatti in presenza di un setpoint nullo, il controllo assumerà il valore preimpostato di 20°C, in modo da permettere una successiva impostazione dei parametri termici, senza la necessità di dover prima raffreddare il sistema.

Di seguito viene riportato il nuovo codice implementato:

**\*\*\*\* FUNZIONE CHE IMPLEMENTA UN CONTROLLO DI TEMPERATURA \*\*\*\***

```
MisuraTemperatura();
tempoPrecedente = tempo;
tempo = millis() * 0.001;
dt = tempo - tempoPrecedente;
LampeggioLed();
LeggiSeriale();
for (int i = 0; i < 3; i++)
{
ControlloPid(i);
}
if (temperatura < valoreTempImp - semiAmpiezzaIsteresi)
{
setPointResImp = 0.3;
tensioneCtrlVent = 0;
setPointRes = setPointResImp;
} else {
if (temperatura >= valoreTempImp + semiAmpiezzaIsteresi)
{
setPointRes = 0;
tensioneCtrlCalcolataRes_1 = 4.2;
erroreRes_1 = 0.1;
erroreIntegraleRes_1 = 0;
erroreDerivatoRes_1 = 0;
primociclo_Res_1 = 1;
tempoPrecedente_1 = 0;
correzioneRes_1 = 0;
CalcolaPWMRes_1();
ImpostaPWM_Risc(resistenza_1);
tensioneCtrlCalcolataRes_2 = 4.2;
erroreRes_2 = 0.1;
primociclo_Res_2 = 1;
erroreIntegraleRes_2 = 0;
erroreDerivatoRes_2 = 0;
tempoPrecedente_2 = 0;
```

```

    correzioneRes_2 = 0;
    CalcolaPWMRes_2();
    ImpostaPWM_Risc(resistenza_2);
    tensioneCtrlVent = 4.2;
    }
}
ComandoVentola();
ControlloResistenze();

```

### 3.8 – Comandi per attivare le operazioni di misura e settaggio dei parametri

Una volta inserite nel firmware le nuove funzioni destinate ad effettuare misure ed impostare parametri elettrici e termici, lo step finale da eseguire riguarda il metodo e i comandi da digitare al terminale per ottenere a video i valori dei parametri richiesti. Assegnando specifici caratteri e numeri alle funzioni, è possibile scegliere la tipologia di funzione da far eseguire al circuito, assegnando al terminale le lettere corrispondenti.

È stato scelto un gruppo di tre lettere per specificare la tipologia di comando e impostazione di un setpoint specifico.

Secondo la posizione in cui è individuato il carattere digitato, si attiva uno specifico comando:

Primo carattere digitato:

- lettera “ m ” individua la richiesta di una funzione che esegue una misura;
- lettera “ s ” individua la funzione che impone un setpoint per uno specifico parametro;

Secondo carattere digitato:

- per le operazioni di misura, il secondo carattere individua il parametro di cui si vuole conoscere il valore:

- “ c “ = Corrente erogata al LED;
- “ f “ = Frequenza del segnale luminoso emesso dal LED;
- “ d “ =Tensione erogata dall'alimentatore del LED;
- “ v “ =Tensione ai capi della serie di LED;
- “ s “ = Tensione ai capi della resistenza di shunt del LED;
- “ t “ = Tensione presente al pin CTRL dell'alimentatore del LED;
- “ t “ seguita da “ e “ = individua la misura di temperatura del sistema;

- per l'impostazione del setpoint, il secondo carattere è sempre la lettera “ p ” ;

Il terzo carattere digitato, individua il colore del rispettivo gruppo di LED del quale si desiderano conoscere i parametri, il riscaldatore e la temperatura :



- “ r “ = individua i LED rossi;
- “ g “ = individua i LED verdi;
- “ b “ = individua i LED blu;
- “ I “ = individua il resistore 1;
- “ s “ = individua il resistore 2;
- “ e “ = individua la richiesta di temperatura;
- “ t “ = tutti i valori richiesti per i tre gruppi di LED;

Di seguito in Fig.3.8, è riportato il menù dei comandi che viene stampato a video sul terminale:

```

SET POINT DI CORRENTE A LED E RISCALDATORI
-----
Sostituire ad (xxxx) un valore numerico
spxxxxx -      compreso tra [0 - 1000]      | r = rosso |
spxxxxx -      0 = 0 mA                    | g = verde |
spxxxxx -      1000 = 0.700 mA             | b = blu   |
spxxxxx -      Per la temperatura (xx)=[0-99]
sptxx   -                                          | t = temp. |
-----

COMANDI PER EFFETTUARE LE MISURE

| m = misura | | x = comando | | t = tutti |
-----

| c | Corrente erogata al led
| f | Frequenza
mxr   x = | d | Tensione erogata dall'alimentatore del led
mxg   x = | v | Tensione ai capi della serie di led (*)
mxb   x = | s | Tensione ai capi della Rshunt del led (*)
mxt   x = | t | Tensione al ctrl dell'alimentatore del led (*)

(*) = Valori ottenuti eseguendo un calcolo

mci = Corrente erogata al riscaldatore 1
mcs = Corrente erogata al riscaldatore 2
mte = Temperatura della piastra

```

Fig.3.8 – Menù dei comandi stampati a video dal terminale

I caratteri impostati nel modo descritto prima, attivano le funzioni richieste. In Tab.3.8 sono riportati i caratteri codificati in codice ASCII, utilizzati all'interno del firmware, e i codici implementati per richiedere espressamente la misura :

CARATTERE	CODIFICA IN CODICE ASCII
t	116
r	114
g	103
b	98
e	101
d	100
s	115
v	118

Tab.3.8 – Caratteri e codifica ASCII

**\*\*\*\*\* STAMPA DEL VALORE DI TENSIONE CALCOLATO AL PIN CTRL DELL'ALIMENTATORE DEI LED \*\*\*\*\***

```

if (leggiStringa[1] == 116) // lettera " t " in seconda posizione
{
if (leggiStringa[2] == 114) // lettera " r " in terza posizione
{
Serial.print(tensioneCtrlCalcolata[rosso], 4); Serial.print("\r");
}
if (leggiStringa[2] == 103) // lettera " g " in terza posizione
{
Serial.print(tensioneCtrlCalcolata[verde], 4); Serial.print("\r");
}
if (leggiStringa[2] == 98) // lettera " b " in terza posizione
{
Serial.print(tensioneCtrlCalcolata[blu], 4); Serial.print("\r");
}
if (leggiStringa[2] == 116) // lettera " t " in terza posizione
{
for (int j = 0; j < 3; j++)
{
Serial.print(tensioneCtrlCalcolata[j], 4); Serial.print("\r");
}
}

```

**\*\*\*\*\* STAMPA IL VALORE DELLA TEMPERATURA MISURATA \*\*\*\*\***

```

if (leggiStringa[1] == 116) // lettera " t " in seconda posizione
{
if (leggiStringa[2] == 101) // lettera " e " in terza posizione
{

```

```

Serial.print(temperatura); Serial.print("\r");
  //Serial.print(numero);Serial.print("\r");
}

```

**\*\*\*\*\* STAMPA MISURA DI TENSIONE EROGATE DAGLI ALIMENTATORI \*\*\*\*\***

```

if (leggiStringa[1] == 100) // lettera " d " in seconda posizione
{
if (leggiStringa[2] == 114) // lettera " r " in terza posizione
{
MisuraTensione(rosso);
Serial.print(tensioneAlimentatoriMisurata[rosso], 4); Serial.print("\r");
}
if (leggiStringa[2] == 103) // lettera " g " in terza posizione
{
MisuraTensione(verde);
Serial.print(tensioneAlimentatoriMisurata[verde], 4); Serial.print("\r");
}
if (leggiStringa[2] == 98) // lettera " b " in terza posizione
{
MisuraTensione(blu);
Serial.print(tensioneAlimentatoriMisurata[blu], 4); Serial.print("\r");
}
if (leggiStringa[2] == 116) // lettera " t " in terza posizione
{
for (int j = 0; j < 3; j++)
{
MisuraTensione(j);
Serial.print(tensioneAlimentatoriMisurata[j], 4); Serial.print("\r");
}
}
}

```

**\*\*STAMPA LE CADUTE DI TENSIONE MISURATE SUI RESISTORI DI SHUNT \*\***

```

if (leggiStringa[1] == 115) // lettera " s " in seconda posizione
{
if (leggiStringa[2] == 114) // lettera " r " in terza posizione
{
MisuraTensioneShunt(rosso);
Serial.print(tensioneShuntCalcolata[rosso], 4); Serial.print("\r");
}
if (leggiStringa[2] == 103) // lettera " g " in terza posizione
{
MisuraTensioneShunt(verde);
Serial.print(tensioneShuntCalcolata[verde], 4); Serial.print("\r");
}
}

```

```

    }
    if (leggiStringa[2] == 98)    // lettera " b " in terza posizione
    {
        MisuraTensioneShunt(blu);
        Serial.print(tensioneShuntCalcolata[blu], 4); Serial.print("\r");
    }
    if (leggiStringa[2] == 116)  // lettera " t " in terza posizione
    {
        for (int j = 0; j < 3; j++)
        {
            MisuraTensioneShunt(j);
            Serial.print(tensioneShuntCalcolata[j], 4); Serial.print("\r");
        }
    }

```

**\*\*\*\*\* STAMPA LA TENSIONE MISURATA AI CAPI DEI LED \*\*\*\*\***

```

    if (leggiStringa[1] == 118)  // lettera " v " in seconda posizione
    {
        if (leggiStringa[2] == 114)  // lettera " r " in terza posizione
        {
            MisuraTensioneLed(rosso);
            Serial.print(tensioneLedCalcolata[rosso], 4); Serial.print("\r");
        }
        if (leggiStringa[2] == 103)  // lettera " g " in terza posizione
        {
            MisuraTensioneLed(verde);
            Serial.print(tensioneLedCalcolata[verde], 4); Serial.print("\r");
        }
        if (leggiStringa[2] == 98) // lettera " b " in terza posizione
        {
            MisuraTensioneLed(blu);
            Serial.print(tensioneLedCalcolata[blu], 4); Serial.print("\r");
        }
        if (leggiStringa[2] == 116)  // lettera " t " in terza posizione
        {
            for (int j = 0; j < 3; j++)
            {
                MisuraTensioneLed(j);
                Serial.print(tensioneLedCalcolata[j], 4); Serial.print("\r");
            }
        }
    }

```

## CAPITOLO 4 – MISURA DELLA TEMPERATURA DI GIUNZIONE DEI LED

### 4.1 – Struttura e principio di funzionamento di un LED

Un diodo LED, è un dispositivo elettronico in grado di permettere il passaggio di corrente in un solo verso. È composto da due terminali, un polo positivo denominato *Anodo* e uno negativo chiamato *Catodo*. Applicando un il livello di tensione ai due terminali, siamo in grado di comandare il diodo e dunque la conduzione di corrente. Esso è composto nella maggior parte dei casi, da materiale semiconduttore drogato con un determinato livello di impurità, per determinare una giunzione denominata P-N, positiva - negativa. Un atomo, possiede un certo numero di elettroni che orbitano attorno ad esso, determinati dal numero atomico dell'elemento, disposti su *Bande* o *Livelli Energetici*, caratterizzati ognuno da un livello energetico. Le bande sono disposte secondo un ordine crescente man mano che la distanza dal nucleo aumenta. Il primo livello energetico contiene due elettroni, il secondo otto mentre l'ultima banda di energia chiamata *Banda di Valenza*, contiene elettroni con livello di energia non sufficientemente elevato da permette loro l'abbandono dell'atomo. Oltre la banda di valenza è presente una seconda banda individuata con il nome di *Banda di Conduzione*, contenente elettroni a un livello di energia maggiore da permettere il distacco dall'atomo, dando luogo al fenomeno della conduzione elettrica. La distanza espressa in energia da fornire all'atomo per permettere a un elettrone di passare da una banda all'altra, definisce la distanza tra le due bande stesse. Questa distanza corrispondente al livello di energia da fornire all'atomo prende il nome di *Band Gap*. Per materiali isolanti, il Band Gap richiede un'energia elevata, nei materiali semiconduttori la distanza tra le due bande è media, mentre per materiali conduttori, la Banda di Valenza e di Conduzione sono sovrapposte, quindi l'energia richiesta per effettuare il passaggio è minima[16].

In Fig.4.1 è riportato lo schema raffigurante la Banda di Valenza, di Conduzione e il Band Gap:

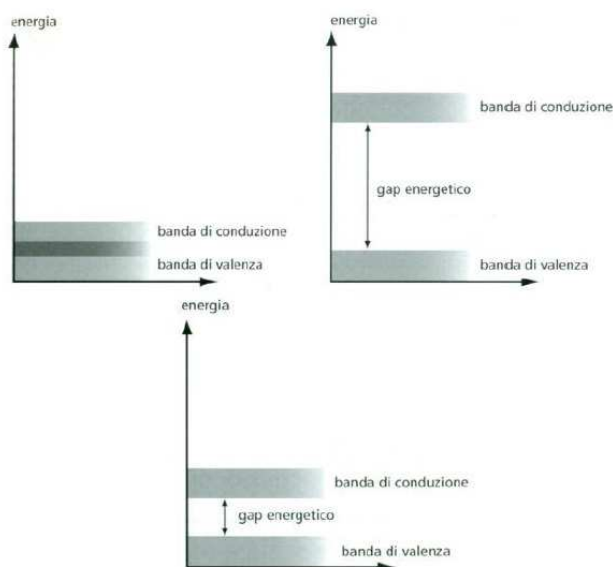


Fig.4.1 – Bande di Valenza, Conduzione e Band Gap[16]

#### 4.2 – Drogaggio di un semiconduttore e la giunzione P-N

Per fare in modo che un semiconduttore possa condurre corrente, è necessario effettuare un'operazione di *Drogaggio* del materiale. Tramite quest'operazione, all'interno del reticolo cristallino contenente gli atomi, è introdotto un elemento atomico contenente un numero maggiore di elettroni rispetto al materiale da drogare, oppure un elemento contenente un numero inferiore di elettroni rispetto al materiale da drogare. Infatti con un elemento pentavalente all'interno del semiconduttore, quattro dei suoi cinque elettroni vengono impiegati per creare il legame covalente tra atomi, il quinto elettrone resta libero e non legato. Questa operazione trasforma il semiconduttore in un materiale drogato di *Tipo N*, perché il materiale acquisisce carica negativa per via degli elettroni in eccesso, trasformando il materiale da semiconduttore a conduttore. Se invece il processo viene eseguito inserendo una sostanza avente un numero di elettroni pari a tre, in numero ridotto rispetto a quelli presenti nel materiale semiconduttore, i tre elettroni dell'atomo vengono impiegati per determinare il legame covalente, ed una lacuna verrà lasciata libera di essere colmata da un elettrone, che a sua volta lascerà libera una lacuna in un'altra zona del reticolo. L'operazione trasforma il semiconduttore drogato di *Tipo P*, e la lacuna può essere considerata una carica positiva che dà luogo ad una corrente elettrica[16].

Ottenute le due tipologie di semiconduttore drogato, mettendole a contatto per ottenere la giunzione P-N, si verifica uno spostamento di cariche. Gli elettroni e le lacune, situati in prossimità della giunzione, neutralizzano il potenziale per via della carica posseduta di segno opposto. Verrà generata dunque una zona priva di cariche polarizzate denominata *Zona di Svuotamento*. Questa zona tenderà a restare vuota, poiché le cariche fisse, determinano una differenza di potenziale in grado di respingere le lacune e gli elettroni, contenute nelle due zone oltre a quella di svuotamento. La differenza di potenziale, espressa con  $V_s$ , tende dunque a creare una barriera di potenziale ed impedire il trasferimento delle cariche. Per consentire al flusso di elettroni di attraversare la barriera, caratterizzata proprio dalla *tensione di soglia*  $V_s$ , dovrà essere applicata agli estremi della giunzione, una tensione maggiore di quella di soglia.

In Fig.4.2 è riportato un particolare della giunzione P-N e della zona di Svuotamento:

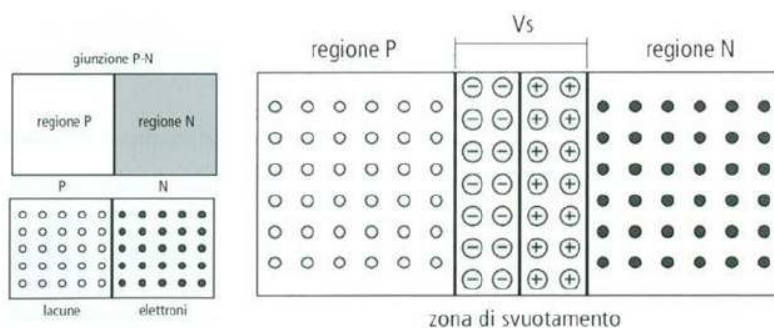


Fig.4.2 – Giunzione P-N (a) – Zona di Svuotamento (b)[16]

#### 4.3 – Polarizzazione della giunzione P-N ed emissione di fotoni

Polarizzare una giunzione P-N significa andare ad applicare un certo livello di tensione agli estremi della giunzione. Se una carica positiva viene applicata al semiconduttore di tipo P in modo da respingere le lacune dal semiconduttore di tipo N, si otterrà la *Polarizzazione Diretta*, mentre applicando una carica negativa al semiconduttore di tipo N per respingere gli elettroni dal semiconduttore di tipo P, si parlerà di *Polarizzazione Inversa*. Lo scopo della polarizzazione è di cercare di comprimere gli elettroni e le lacune il più possibile in prossimità della giunzione, in modo da ridurre la Zona di Svuotamento, e aumentare il numero di elettroni in grado di attraversare la barriera di potenziale aumentando il voltaggio e dunque la corrente. Quando un elettrone attraversa la barriera ricombinandosi con una lacuna, esso decade a un livello di energia minore, rilasciando l'energia posseduta in eccesso sotto forma di *Fotone* e quindi radiazione luminosa. La radiazione luminosa emessa avrà una lunghezza d'onda e quindi un colore dipendente dalla distanza presente tra i due livelli energetici dei materiali utilizzati per costituire la giunzione P-N [16].

In Fig.4.3 è riportato un esempio esplicativo:

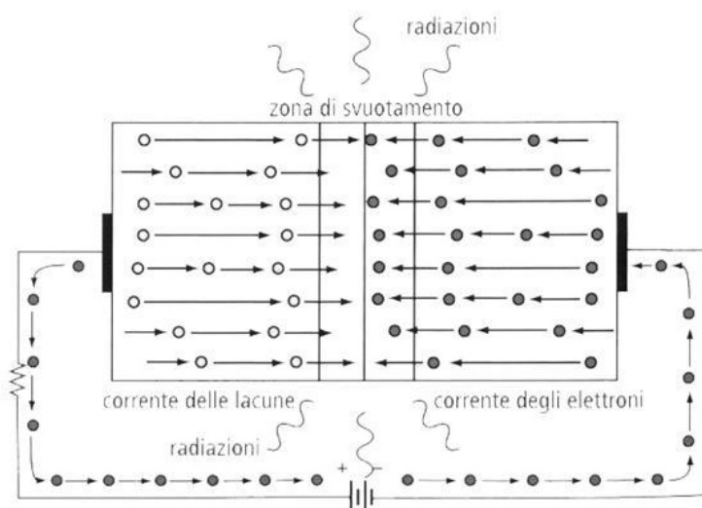


Fig.4.3 – Polarizzazione ed emissione di fotoni [16]

Un diodo LED in grado di produrre una radiazione luminosa, si differenzia dalle comuni lampade a incandescenza e dai comuni tubi fluorescenti, poiché è in grado di fornire una radiazione luminosa monocromatica, cioè una radiazione avente una determinata lunghezza d'onda nello spettro di emissione della luce. Come espresso nei paragrafi precedenti, la lunghezza d'onda della radiazione dipende principalmente dal materiale utilizzato per creare la giunzione P-N, e dal livello di drogaggio ai quali sono stati sottoposti i materiali semiconduttori utilizzati per formare la giunzione.

In Fig.4.3.1 è riportato l'andamento della distribuzione di potenza spettrale relativa a vari LED presi in esame:

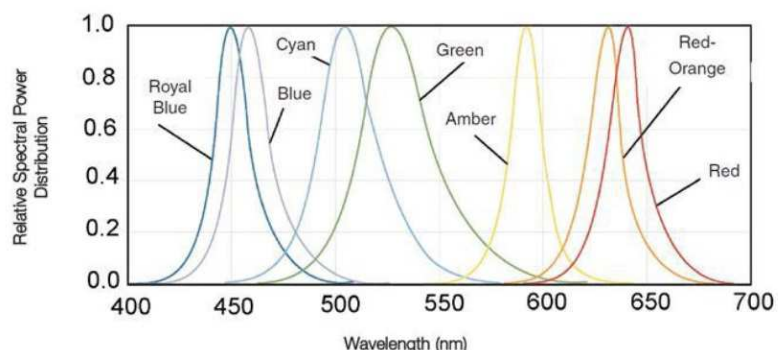


Fig.4.3.1 – Andamenti di distribuzione spettrale [16]

Per osservare la differente lunghezza d'onda prodotta da diodi LED composti da diversi materiali, in Tab.4.3 è riportato un resoconto con differenti materiali utilizzati:

Lunghezza d'onda		Voltaggio di andata		Angolo di vista	Materiale del chip
nm	Nome del colore	20mA	Intensità per LED da 5mm	(gradi)	
940	IR	1.7	16mW@50mA	15	GaAlAs/GaAs
880	IR	1.7	18mW@50mA	15	GaAlAs/GaAs
850	IR	1.8	26mW@50mA	15	GaAlAs/GaAs
660	Ultra rosso	2.2	2000mcd@50mA	15	GaAlAs/GaAs
635	Rosso ad alta efficienza	2.2	200mcd@20mA	15	GaAsP/GaP
633	Superrosso	2.2	3500mcd@20mA	15	InGaAlP
623	Rosso arancio	2.1	4500mcd@20mA	15	InGaAlP
620	Sopra arancio	2.2	4500mcd@20mA	15	InGaAlP
612	Arancio	2.1	160mcd@20mA	15	GaAsP/GaP
595	Sopragiallo	2.1	5500mcd@20mA	15	InGaAlP
592	Giallo ambra	3.6	7000mcd@20mA	15	InGaAlP
585	Giallo	3.6	100mcd@20mA	15	GaAsP/GaP
3500K	Bianco "incandescente"	2.2	200mcd@20mA	20	SiC/GaN
4500K	Bianco "incandescente"1.5	2.2	2000mcd@20mA	20	SiC/GaN
5000K	Bianco pallido	3.6	4000mcd@20mA	20	SiC/GaN
6500K	Bianco pallido	3.6	4000mcd@20mA	20	SiC/GaN
8000K	Bianco freddo	3.6	6000mcd@20mA	20	SiC/GaN
574	Giallo limone super	2.4	1000mcd@20mA	15	InGaAlP
570	Verde limone super	2.0	1000mcd@20mA	15	InGaAlP
565	Verde ad alta efficienza	2.1	200mcd@20mA	15	GaP
560	Verde super puro	2.1	350mcd@20mA	15	InGaAlP
555	Verde puro	2.1	80mcd@20mA	15	GaP
525	Verde acqua	3.5	10000mcd@20mA	15	SiC/GaN
505	Verde blu	3.5	2000mcd@20mA	15	SiC/GaN
470	Super blu	3.6	3000mcd@20mA	15	SiC/GaN
430	Ultra blu	3.8	100mcd@20mA	15	SiC/GaN

Tab.4.3 – Materiali utilizzati per la giunzione P-N [16]

#### 4.4 – Power LED

Come già ampiamente descritto nei capitoli precedenti, per lo sviluppo di questo progetto di tesi, sono stati presi in esame i Power LED della LUXEON, per le caratteristiche elettriche e



illuminotecniche fornite da questa famiglia di LED. Essi, nati circa nove anni fa, possiedono tutte le caratteristiche tecniche per essere utilizzati in ambienti interni ed esterni nel settore civile e in quello industriale. Sono caratterizzati da una struttura costruttiva piatta, montati con la parte inferiore su un dissipatore di calore e con la possibilità di collegare i terminali di alimentazione lateralmente. Nella parte inferiore del LED è presente una placca di alluminio liscia perfettamente levigata, studiata per essere lei stessa un dissipatore di calore per il LED, e studiata per essere collegata ad un dissipatore più ampio per l'asportazione del calore.

In Fig.4.4 è riportato un Power LED come quelli presi in esame:



*Fig.4.4 – Power LED[16]*

A sinistra in Fig.4.4 è riportato il LED, a destra è riportato il LED connesso al dispositivo dissipatore di calore.

La produzione di luce ottenuta da componenti allo stato solido, viene generata attraverso fenomeni fisici direttamente collegati alla temperatura di giunzione raggiunta durante la ricombinazione degli elementi.

#### 4.5 – La temperatura di giunzione $T_j$

La temperatura di giunzione, [16] come spesso erroneamente viene interpretata, non è la temperatura misurata tra la capsula contenente la vera giunzione del LED ed il supporto sul quale la giunzione è stata montata per permettere la dissipazione del calore. Infatti molto spesso la temperatura di giunzione, viene interpretata come quella misurata attraverso l'utilizzo di una termocoppia o laser sul supporto complessivo costituente il LED. La temperatura misurata attraverso una termocoppia è in realtà composta dalla temperatura raggiunta dai vari componenti costituenti la capsula. Per analizzare questi elementi, sono considerati attraverso resistenze analoghe a quelle elettriche, i vari contributi di calore dei singoli componenti. In Fig.4.5 è riportato lo schema descrittivo delle resistenze termiche:

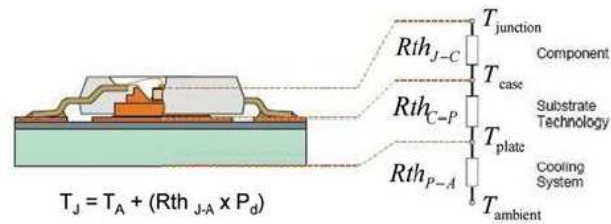


Fig.4.5 – Resistenze termiche [16]

Come si può notare dalla Fig.4.5 sono presenti tre resistenze termiche, collegate in serie tra loro, ognuna riconducibile ad un contributo di temperatura specifico di un settore del LED completo:

- $R_{th\ J-C}$  : Resistenza termica tra la giunzione e il case di contenimento;
- $R_{th\ C-P}$ : Resistenza termica tra il case e il piano di appoggio;
- $R_{th\ P-A}$  : Resistenza termica tra il piano di appoggio e l'ambiente;

Considerando la struttura del LED formata in modo analogo a un semplice circuito elettrico, il calore che attraversa la resistenza termica, passando da un punto a temperatura maggiore verso uno a temperatura minore, genera al suo passaggio un salto termico in ogni singola resistenza termica, dato dal prodotto tra il calore in transito e il valore della resistenza termica costante. La conduzione termica dunque lega il degrado prestazionale della tensione alla temperatura raggiunta dal sistema.

In Fig.4.5.1 è riportata la caratteristica che lega la tensione alla temperatura di giunzione, per differenti valori di corrente:

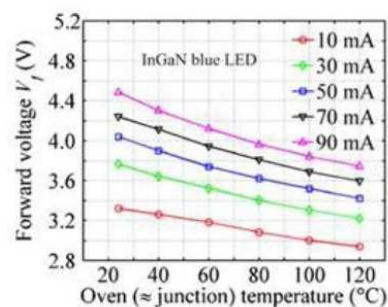


Fig.4.5.1 – Caratteristica Tensione-Temperatura di giunzione [16]

Per osservare più da vicino i componenti costituenti il LED, la struttura può essere scomposta nei suoi quattro elementi principali:

- *GIUNZIONE*: contenente tutto il chip del LED ;
- *CAPSULA*: contenente i filtri, la colla e i fosfori;

- *PIASTRA DI RAFFREDDAMENTO* (Head Slug): contenente la piastra per la conduzione di calore e il riflettore ottico montato sotto la giunzione;
- *COOLER* (Head Sink): cioè il dissipatore di calore vero e proprio attaccato alla piastra;

In Fig.4.5.2 vengono evidenziati i quattro componenti descritti in precedenza:

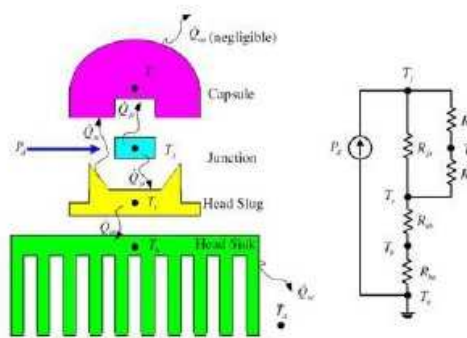


Fig.4.5.2 – Componenti di un LED [16]

#### 4.6 – Misura della temperatura di giunzione $T_j$

La temperatura di giunzione del LED influisce su un notevole numero di parametri elettrici ed ottici, come la lunghezza d'onda di picco dal quale dipende il colore, la lunghezza della banda di emissione e il degrado dei componenti e quindi la durata di vita del LED.

Per effettuare la misura della temperatura di giunzione sono note varie tecniche, ma indubbiamente la più utilizzata consiste nella misura della tensione in ingresso al LED. Per effettuare questa misura, i LED vengono inseriti all'interno di un forno con pareti isolate, il quale permette di controllare il livello di temperatura al suo interno. La giunzione del LED inserita nel forno, una volta raggiunto l'equilibrio termico, tenderà ad assumere la stessa temperatura interna al forno la quale verrà fatta aumentare gradualmente da un valore iniziale di 20°C fino al valore di 120°C mentre contemporaneamente verrà misurata la tensione presente ai capi del LED a diversi livelli di corrente impressa[18]. In questo modo siamo in grado di stimare come la temperatura raggiunta dalla giunzione, influisce su tutti i parametri fondamentali di emissione del LED stesso.

Prendendo in considerazione l'equazione che lega la temperatura di giunzione  $T_j$  con il flusso luminoso emesso si ha che:

$$\varphi_2 = \varphi_1 e^{-k_t(T_2-T_1)}$$

Dove con  $\varphi_1$  e  $\varphi_2$  s'indicano i flussi luminosi emessi a due differenti valori di temperatura di giunzione  $T_1$  e  $T_2$  con ( $T_2 > T_1$ ) e  $k_t$  è il coefficiente di temperatura dei LED.

Dall'equazione precedente, si nota il legame tra l'elevata temperatura di giunzione e la degradazione del flusso luminoso emesso. Attraverso la relazione che lega il flusso luminoso al tempo di funzionamento riportata di seguito, s'individua come la riduzione del flusso diminuisce in maniera esponenziale:

$$\varphi = \varphi_0 e^{-\beta t}$$

dove con  $\varphi_0$  si indica il flusso luminoso inizialmente emesso,  $\beta$  è il livello di degradazione, e  $t$  è il tempo di funzionamento.

Il livello di degradazione  $\beta$  viene espresso attraverso l'equazione:

$$\beta = \beta_0 I_F e^{-\frac{E_a}{kT_j}}$$

con  $\beta_0$  viene indicato il livello di degradazione comune a tutti i LED,  $I_F$  è la corrente di funzionamento,  $E_a$  è l'energia di attivazione,  $k$  è la costante di Boltzmann e  $T_j$  la temperatura di giunzione.

Individuato il legame tra flusso luminoso emesso e la temperatura di giunzione, la funzione che lega la temperatura di giunzione e la tensione applicata alla serie di LED, mantenendo costante la corrente di funzionamento  $I_F$ , può essere assunta lineare con normali condizioni di funzionamento e per una temperatura di giunzione di circa 100°C, mentre viene assunta non lineare per una temperatura di giunzione superiore a 100°C. La relazione matematica che lega le due grandezze è riportata di seguito:

$$T_j = m * V_f + T_0$$

dove con  $m = \frac{\Delta T}{\Delta V}$  espressa in °C/V ottenuta dalla caratteristica ( $V_f - T_j$ ) riportata in Fig.4.6, e  $T_0$  è l'intercetta sull'asse della temperatura.

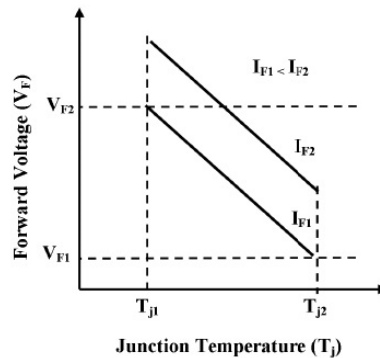


Fig.4.6 – Caratteristica Tensione – Temperatura – Corrente [18]

Tramite la misura di due livelli di tensione  $V_{F1}$  e  $V_{F2}$  eseguite a due differenti livelli di temperatura di giunzione  $T_{j1}$  e  $T_{j2}$  mantenendo la corrente di funzionamento costante  $I_{F1}$ , la pendenza e l'intercetta in  $T_0$ , permettono di ottenere la relazione per i LED della corrente di funzionamento  $I_{F1}$ . Dalla caratteristica  $V_f - T_j$  la relazione è lineare con corrente di funzionamento costante, quindi tramite la misura di tensione  $V_F$  con una corrente di funzionamento  $I_F$  fornita, si può stimare la temperatura di giunzione finale  $T_j$ .

#### 4.7 – Stima della temperatura di giunzione con il nuovo sistema elettrico

Dopo aver implementato le potenzialità possedute dal nuovo circuito elettrico, il risultato complessivo ha permesso di effettuare un'ulteriore misura molto più complessa ed articolata, rispetto le precedenti misure richieste. Il circuito è la base dal quale partire per poter stimare con buona approssimazione un valore di temperatura, ma non di meno risulta essere il codice scritto attraverso il software di calcolo MatLab, che permette di gestire ed attivare il circuito e i comandi necessari in modo automatico. Infatti, questa particolare misura, richiede una sequenza di comandi in serie, sviluppati in un tempo ridottissimo in maniera automatica. La sequenza di operazioni da eseguire risulta essere:

1. Imposto la temperatura del sistema;
2. Attraverso il controllo si mantiene costante la temperatura;
3. Misuro la temperatura inizialmente raggiunta dal sistema;
4. Impongo la corrente richiesta ai LED;
5. Misuro la corrente erogata ai LED;
6. Misuro la tensione applicata alla serie di LED;
7. Annulla la corrente erogata ai LED;
8. Misuro la temperatura finale del sistema;

Le otto operazioni citate, vengono eseguite in un tempo massimo stimato di circa 11sec per ogni LED. Infatti in modo automatico, queste operazioni vengono eseguite per ogni gruppo di LED ciclicamente secondo uno schema ben preciso. All'interno del codice implementato

in MatLab, è stato inserito un ciclo che permette di effettuare un aumento di corrente graduale, a partire da un valore minimo di setpoint imposto ai LED pari a 0.200mA fino al massimo erogabile di 0.650 mA con aumenti di corrente distribuiti in 3 scatti di circa 0.200 mA ciascuno. Attraverso questa sequenza di operazioni eseguite per ogni gruppo di LED, siamo in grado di stimare una caratteristica che lega la tensione erogata ai capi dei LED, in funzione della corrente erogata costante a un certo valore imposto di temperatura. In questo modo la caratteristica assume un andamento esponenziale per differenti valori di tensione, corrente e temperatura misura.

In Fig.4.7 è riportato un esempio di caratteristica ottenuta per una stringa di LED alimentata da una tensione crescente a partire da 130 V e una corrente erogata a partire da 0 A.

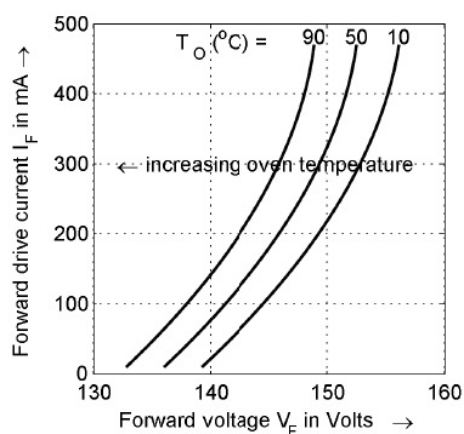


Fig.4.7 – Caratteristica Tensione – Corrente [18]

#### 4.8 – Codice MatLab per la misura di temperatura di giunzione $T_j$

Il codice MatLab [19] scritto per far eseguire le istruzioni ad Arduino e di conseguenza al circuito elettrico, è composto da una sequenza di istruzioni, che attivano comandi presenti in determinati file di testo scritti precedentemente. Inoltre è stato scelto di introdurre intervalli di circa otto secondi nel ciclo di misura per ogni LED, in modo tale che l'accensione non comporti l'immissione di calore nel LED tale da modificare in modo significativo la temperatura di giunzione, rispetto a quella impostata e controllata dal sistema e misurata con la sonda.

MatLab sfrutta il codice già presente in Arduino per comandare il funzionamento generale. Attraverso i file di testo, sono scritte piccole operazioni come l'apertura della porta seriale, la lettura del buffer, l'impostazione di un setpoint di corrente, la chiusura della porta seriale, o comunque tutte le operazioni che il controllore ha già al suo interno. I file di testo permettono di attivare la porzione di codice interessata.

Gli script maggiormente usati sono:

- ApriSeriale();
- ScriviSeriale2();

- LeggiSeriale();
- ChiudiSeriale();

Questi file di testo, attraverso l'immissione del comando tra parentesi della misura che si vuole andare ad effettuare, si comportano esattamente come il terminale TeraTerm utilizzato per imporre i comandi dall'esterno ad Arduino. Il codice MatLab invece, permette di richiamare ogni singolo file con l'istruzione voluta già inserita, ed esegue le istruzioni automaticamente, senza dover settare i parametri esternamente. Di seguito, sono riportati i codici presenti nei file di testo descritti in precedenza e il codice che esegue esattamente in sequenza gli otto passaggi descritti nel paragrafo precedente:

#### FUNZIONE ApriSeriale()

```
function [ s1 ] = ApriSeriale()
s1 = serial('COM4', 'BaudRate', 115200, 'DataBits', 8, 'Parity', 'none',
'Terminator', 'CR');
fopen(s1);
[s,i] = LeggiSeriale1( s1 )
end
```

In questa funzione viene impostata la porta di comunicazione tra il terminale ed Arduino, la velocità di scrittura sulla porta seriale, la velocità di lettura e legge i dati presenti nel buffer.

#### FUNZIONE ScriviSeriale2()

```
function ScriviSeriale2( s1, comando );
fprintf (s1 , comando );
end
```

In questa funzione viene passato il nome della variabile dove verranno memorizzati i dati, e viene scritto il comando attraverso fprintf.

#### FUNZIONE LeggiSeriale()

```
function [ value ] = LeggiSeriale( s1 )
[letto,count,msg] = fscanf(s1);
value = str2num(letto);
end
```

Questa funzione mi permette di leggere i dati presenti all'interno del buffer.

#### FUNZIONE ChiudiSeriale()

```
function [] = ChiudiSeriale(s)
fclose(s);
end
```

Attraverso questa funzione, una volta terminate tutte le operazioni desiderate, viene chiuso il collegamento con la porta seriale tra Arduino e il terminale utilizzato.

Tutte queste funzioni, come già espresso, vengono eseguite automaticamente, tramite un codice nel quale sono state inserite e richiamate nel momento più opportuno durante il processo di misura. Di seguito è riportato il codice implementato in MatLab per effettuare la misura:

FUNZIONE Misura\_Tj.m:

```
temp_set = [25:5:35];
N_temp = length(temp_set);
cur_set = [300 700 1000];
N = length(cur_set);
temp_mis_iniz = zeros(3,N,N_temp);
temp_mis_fin = zeros(3,N,N_temp);
cur_mis = zeros(3,N,N_temp);
tens_mis = zeros(3,N,N_temp);
temp_mis = 0;
cond_fin = 1;
delta = 1;
led_color = 'rgb';
x = zeros(1,N_temp);
y = zeros(1,N_temp);

for m = 1:N_temp
    ScriviSeriale2(s1,['spt' num2str(temp_set(m))])
    pause(0.1);
while (cond_fin)
    ScriviSeriale2(s1,'mte');
    temp_mis = LeggiSeriale(s1);
if (temp_mis>temp_set(m) - delta &temp_mis<temp_set(m) + delta)
cond_fin = 0;
end
    pause(2);
end

temp_mis
cond_fin = 1;

% IMPLEMENTO IL METODO DI MISURA PER I LED

for k = 1:N
forn_color = 1:3
    ScriviSeriale2(s1,'mte');
temp_mis_iniz(n_color,k,m) = LeggiSeriale(s1);
    pause(0.1);
if cur_set(k) < 1000
cur_set_str = ['0' num2str(cur_set(k))]
else
cur_set_str = num2str(cur_set(k))
end
    ScriviSeriale2(s1,['sp'led_color(n_color) cur_set_str]);
pause(1);
    ScriviSeriale2(s1,['mc'led_color(n_color)]);
```



```

        pause(0.1);
cur_mis(n_color,k,m) = LeggiSeriale(s1)
        pause(0.1);
        ScriviSeriale2(s1,['mv'led_color(n_color)]);
        pause(0.1);
tens_mis(n_color,k,m) = LeggiSeriale(s1);
        pause(0.1);
        ScriviSeriale2(s1,['sp'led_color(n_color) '0000']);
        pause(0.1);
        ScriviSeriale2(s1,'mte');
        pause(0.1);
temp_mis_fin(n_color,k,m) = LeggiSeriale(s1);
        pause(0.1);
        pause(5);

end
end
end

figure
hold on
for m = 1:N_temp
    plot ([1:N],temp_mis_iniz(:,m))
    plot ([1:N],temp_mis_fin(:,m))
    xlabel('N valori di funzionamento del ciclo')
    ylabel('Temperatura piastra [°C]')
    title('Caratteristica della temperatura iniziale e finale per N
valori di funzionamento del ciclo')
end
hold off

figure
hold on
for m = 1:N_temp
    plot (tens_mis(1,:,m)/2,cur_mis(1,:,m),'r')
    plot (tens_mis(2,:,m)/2,cur_mis(2,:,m),'g')
    plot (tens_mis(3,:,m)/5,cur_mis(3,:,m),'b')
    plot (tens_mis(1,:,m)/2,cur_mis(1,:,m),'r*')
    plot (tens_mis(2,:,m)/2,cur_mis(2,:,m),'g*')
    plot (tens_mis(3,:,m)/5,cur_mis(3,:,m),'b*')
    xlabel('Tensione ai capi di un singolo LED [V]')
    ylabel('Corrente ai capi di un singolo LED [A]')
    title('Caratteristica corrente-tensione di un LED ad una determinata
temperatura')
end
hold off

figure
hold on
for n = 1:3
    for m = 1:N
        for k = 1:N_temp
            x(k) = temp_mis_iniz(n,m,k);
            y(k) = tens_mis(n,m,k)/2;

        end
        plot(x,y,led_color(n))
    end
end

for m = 1:N

```

```

for k = 1:N_temp
    x(k) = temp_mis_fin(n,m,k);
    y(k) = tens_mis(n,m,k)/2;
end
plot(x,y,led_color(n))
end
end

xlabel('Temperatura di Giunzione Tj [ °C ]')
ylabel('Tensione ai capi dei singoli LED [ V ]')
title('Decadimento delle prestazioni del LED in funzione di Tj con
corrente costante')
end
end
holdoff

```

Nella parte terminale, è presente anche una porzione di codice in grado di far stampare a MatLab i valori e il grafico della misura effettuata, per visualizzare direttamente l'andamento della caratteristica fornita dalle operazioni di misura.

#### 4.9 - Risultati ottenuti eseguendo le misure

Il sistema oggetto di studio viene portato prima di tutto ad un livello di temperatura desiderato, e mantenuto nell'intorno di questo valore tramite il controllo di temperatura. Raggiunto il livello termico richiesto, al LED viene fornito una corrente costante e contemporaneamente viene misurata la tensione ai suoi capi. L'operazione con lo stesso valore di corrente e temperatura, viene eseguita anche per i LED verdi e blu. Quando le misure di tensione sono state effettuate per la terna di LED, si passa ad un livello di corrente e temperatura maggiore, e si effettuano nuovamente le misure. Il sistema procede con lo stesso metodo, fino a quando non sono eseguite le misure di tutti i valori richiesti, in modo automatico, per le temperature di 25,30,35 °C e le correnti di 0.200 mA, 0.400 mA e 0.650 mA.

Di seguito sono riportati i valori delle misure effettuate e descritte in precedenza, per diversi valori di temperatura e corrente.

Nelle Tab.4.9 (a-b-c), vengono descritti gli andamenti della tensione misurata ai LED, per differenti valori di corrente a temperatura costante:

T = 25 [°C]	I = 0.200 [mA]	I = 0.400 [mA]	I = 0.700 [mA]
LED ROSSI	4.3523	4.8612	5.1434
LED VERDI	5.8707	6.5031	6.7780
LED BLU	15.1694	16.4255	16.9880

(a)

T = 30 [°C]	I = 0.200 [mA]	I = 0.400 [mA]	I = 0.700 [mA]
LED ROSSI	4.3711	4.8679	5.1230
LED VERDI	5.9168	6.5130	6.7551
LED BLU	15.2136	16.5375	16.9138

(b)

T = 35 [°C]	I = 0.200 [mA]	I = 0.400 [mA]	I = 0.700 [mA]
LED ROSSI	4.3654	4.8579	5.0984
LED VERDI	5.9291	6.5042	6.7258
LED BLU	15.3123	16.5800	16.8360

(c)

Tab. 4.9 – Valori di tensione misurati con temperatura e corrente costante

Come si può notare dai dati ottenuti tramite le misure eseguite con MatLab, i valori di tensione, specialmente quelli misurati con valori di corrente massima, rispecchiano la degradazione della tensione ai capi della serie di LED in funzione dell'aumento di temperatura di giunzione  $T_j$ .

	I = 0.700 [mA] T = 25 [°C]	I = 0.700 [mA] T = 30 [°C]	I = 0.700 [mA] T = 35 [°C]
LED Rossi	5.1434	5.1230	5.0984
LED Verdi	6.7780	6.7551	6.7258
LED Blu	16.9880	16.9138	16.8360

Si nota come all'aumentare della corrente, pur mantenendola costante durante il processo di misura, il decadimento di tensione è maggiore in corrispondenza dei valori più alti che corrente erogata, in accordo con quanto descritto nei paragrafi precedenti.

Per rendere maggiormente visibile i risultati ottenuti, vengono stampati i dati, attraverso l'ausilio di grafici riportati in Fig.4.9:

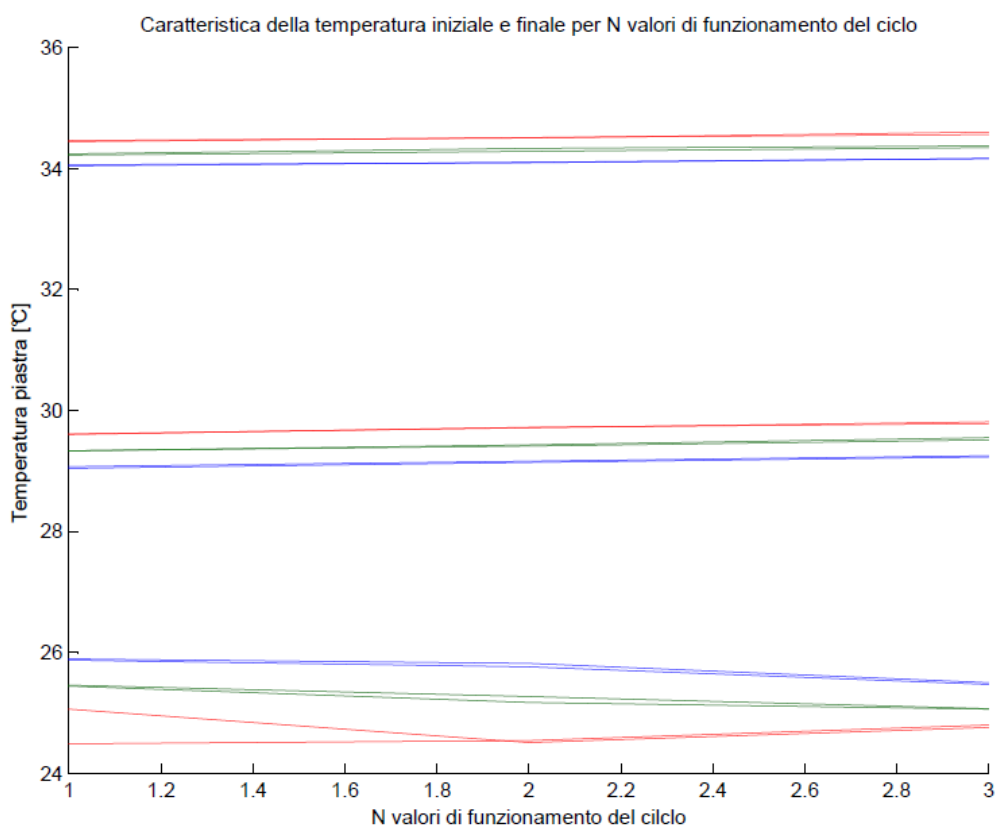


Fig. 4.9 (a) – Caratteristica Temperatura – N cicli

In Fig.4.9 (a) vengono messi in relazione i dati di temperatura ricavati, in funzione del numero di cicli che il software compie. Per ogni valore di temperatura impostato, a partire da 25°C fino al valore massimo di 35°C, vengono misurate le temperature iniziali e finali raggiunte dalla giunzione di ogni singolo LED. Per ogni misura effettuata, si nota che il controllo di temperatura, tende a mantenere costante il valore impostato, azionando i riscaldatori o la ventola permettendo il mantenimento attraverso il controllo, del valore di temperatura nell'intorno del valore impostato. All'avvio del processo di misura, si notano discrepanze dei valori, soprattutto in corrispondenza della misura effettuata sul LED rosso nell'intorno del valore di 25°C, poiché si presume che il supporto metallico contenente i LED, si trova inizialmente ad una temperatura più bassa rispetto la seconda temperatura misura. La differenza tra il valore iniziale di temperatura misurato e il valore finale, è influenzato sia dai valori di soglia inseriti nel ciclo d'isteresi all'interno del controllo, il quale ha il compito di mantenere la temperatura impostata, con un massimo scostamento del  $\pm 0.25$  °C, che dall'errore di  $\pm 0.5$   $\mu\text{A/k}$  introdotto dal trasduttore utilizzato per effettuare la misura.

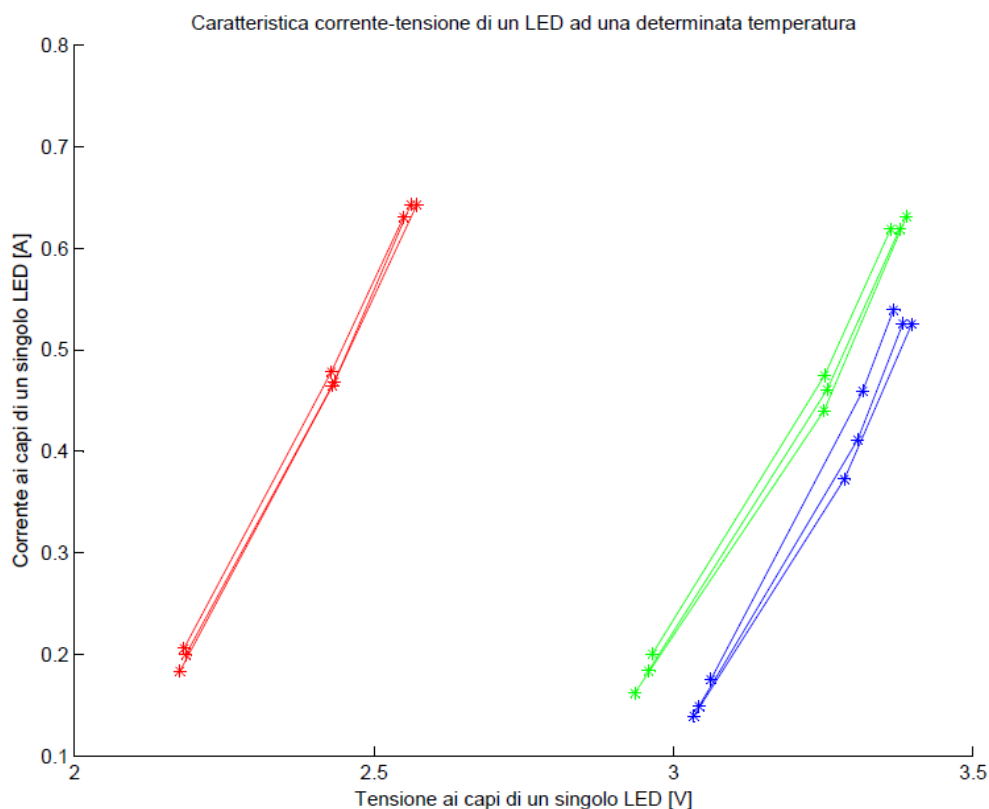


Fig.4.9 (b) – Caratteristica Corrente – Tensione ai capi di un LED con temperatura misurata costante

In Fig.4.9 (b) vengono riportati gli andamenti delle caratteristiche ottenute impostando i tre valori di temperatura ed un valore di corrente, andando a misurare il valore di tensione ai capi di un singolo LED. La prima valutazione di tensione è stata eseguita per una temperatura di giunzione pari a 25°C, la seconda a 30°C e la terza a 35°C con una corrente erogata ai LED di circa 0.200 mA. Per il LED rosso, si nota che i valori di tensione misurati con queste impostazioni di funzionamento del sistema, si riducono in funzione dell'aumento della temperatura di giunzione. Dunque le curve di colore rosso, tendono a traslare da destra verso sinistra portandosi ad un valore di tensione inferiore. La stessa cosa accade per i LED verdi e i LED blu.

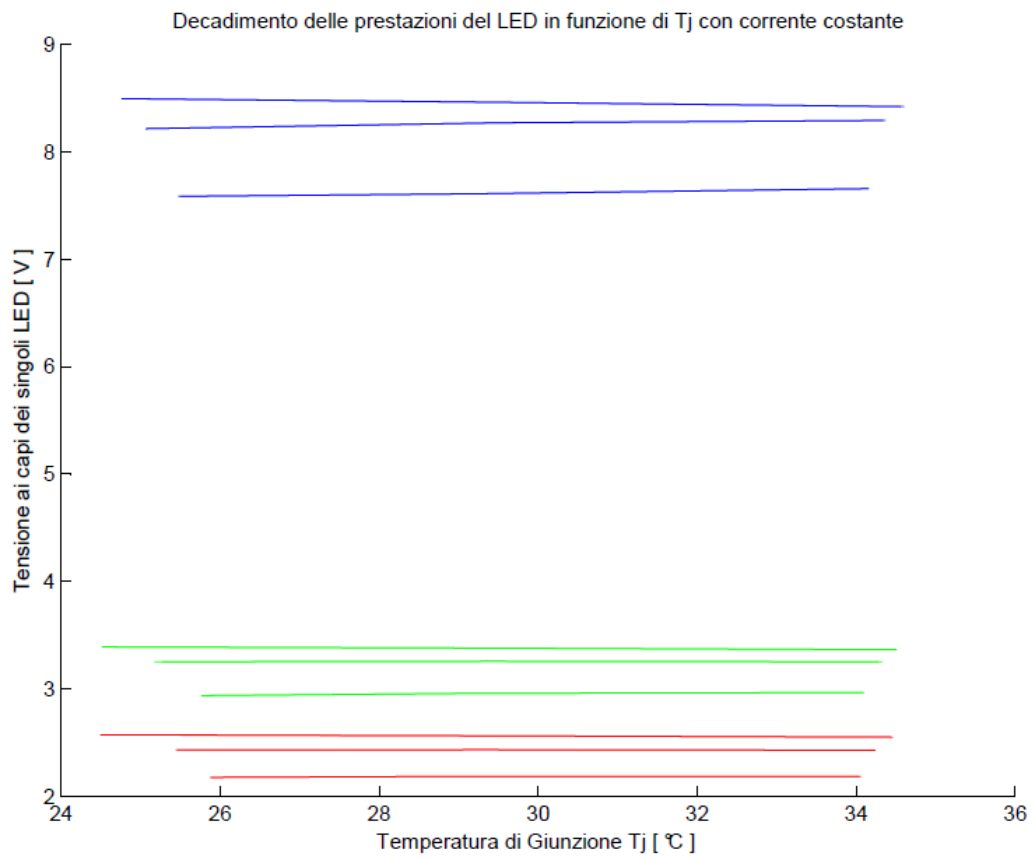


Fig.4.9 (c) – Caratteristica Tensione – Temperatura con corrente erogata costante

Le caratteristiche riportate in Fig.4.9 (c), rappresentano la variazione di tensione, in funzione della temperatura di giunzione  $T_j$  con una corrente erogata al LED costante. A causa della ridotta variazione della temperatura ottenibile tramite il sistema di misura, la caratteristica non evidenzia una pendenza accentuata, ma si può comunque individuare la riduzione di tensione misurata all'aumentare della temperatura di giunzione. Le tre caratteristiche per ogni colore, rappresentano ognuna un funzionamento con un valore di corrente man mano superiore. La caratteristica posta più in basso per ogni gruppo di LED, individua il valore di corrente minore impostato a circa 0.200 mA, mentre l'ultima caratteristica posta più in alto per ogni gruppo di LED, rappresenta il valore di corrente massima erogata al LED ed impostato a circa 0.650 mA.

## *CAPITOLO 5 – CONCLUSIONI E SVILUPPI FUTURI*

### *5.1 - Conclusioni*

In questo lavoro di tesi è stato progettato, realizzato e messo in funzione un nuovo circuito elettrico per la gestione e il controllo dei parametri illuminotecnici, termici ed elettrici di una sorgente a LED con impostazione controllata della cromaticità. Il progetto è stato realizzato e costruito interamente all'interno del laboratorio di Illuminotecnica e Fotometria dell'Università degli Studi di Padova. Il circuito elettrico, studiato inizialmente per porre rimedio ad una precedente soluzione scomoda, si è dimostrato essere un ottimo strumento utile per la calibrazione e la misura di parametri termici del sistema. Infatti attraverso l'ausilio del software scritto ed implementato tramite Arduino e MatLab, è stato possibile imporre una determinata percezione cromatica da visualizzare. Inoltre si può misurare la temperatura di giunzione dei LED, per poter essere in grado di stimare la degradazione, con la possibilità di effettuare in futuro prove di invecchiamento rapido. Dunque tramite questo lavoro di tesi, è stato sviluppato un dispositivo di discreta complessità, in grado di fornire all'utente un numero utile di parametri elettrici ed illuminotecnici.

Alcuni dei parametri che possono essere misurati e valutati attraverso il circuito costruito sono:

- Misura di corrente erogata ai LED;
- Misura di tensione ai capi dei LED;
- Misura di temperatura del sistema;
- Impostazione di una determinata cromaticità ai LED;
- Controllo e mantenimento della temperatura in un determinato range di valori;
- Impostazione di un valore di temperatura di funzionamento;
- Misura della temperatura di giunzione  $T_j$  dei LED;
- Misura di frequenza proporzionale a un segnale luminoso;

### *5.2 – Sviluppi futuri*

Uno degli sviluppi futuri di notevole rilevanza da apportare al circuito, riguarda lo sviluppo di un supporto stampato per il circuito elettrico, in modo da rendere più piccolo, compatto e maneggevole il circuito stesso. Attualmente il circuito è affidabile e robusto, ma la costruzione di una nuova scheda elettronica stampata, aiuta sicuramente ad attenuare sia gli errori che l'affidabilità delle misure effettuate.

Il secondo aspetto da sviluppare riguarda l'implementazione di un'interfaccia di controllo facile e intuitiva, in modo da permettere anche agli utenti meno esperti in linguaggi di

programmazione, di selezionare una determinata percezione cromatica direttamente sulla curva dello spazio dei colori definita dalla CIE.

Con l'implementazione della misura di temperatura di giunzione  $T_j$  del LED, si è resa necessaria la coibentazione ed il miglioramento termico del sistema contenente i LED. Infatti allo stato attuale lo scambio termico della piastra di supporto con l'ambiente circostante, risulta essere fortemente accentuato. Una possibile soluzione, è quella di apporre al sistema una struttura coibentata, in grado di permettere contemporaneamente il raffreddamento del sistema e il mantenimento della temperatura interna.



### *LISTATI MATLAB UTILIZZATI*

**ApriSeriale.m:** consente di aprire la comunicazione tra PC ed Arduino attraverso la porta seriale.

**ChiudiSeriale.m:** consente di chiudere la comunicazione tra PC ed Arduino.

**ScriviSeriale2.m:** consente la sola scrittura del comando sul buffer di Arduino.

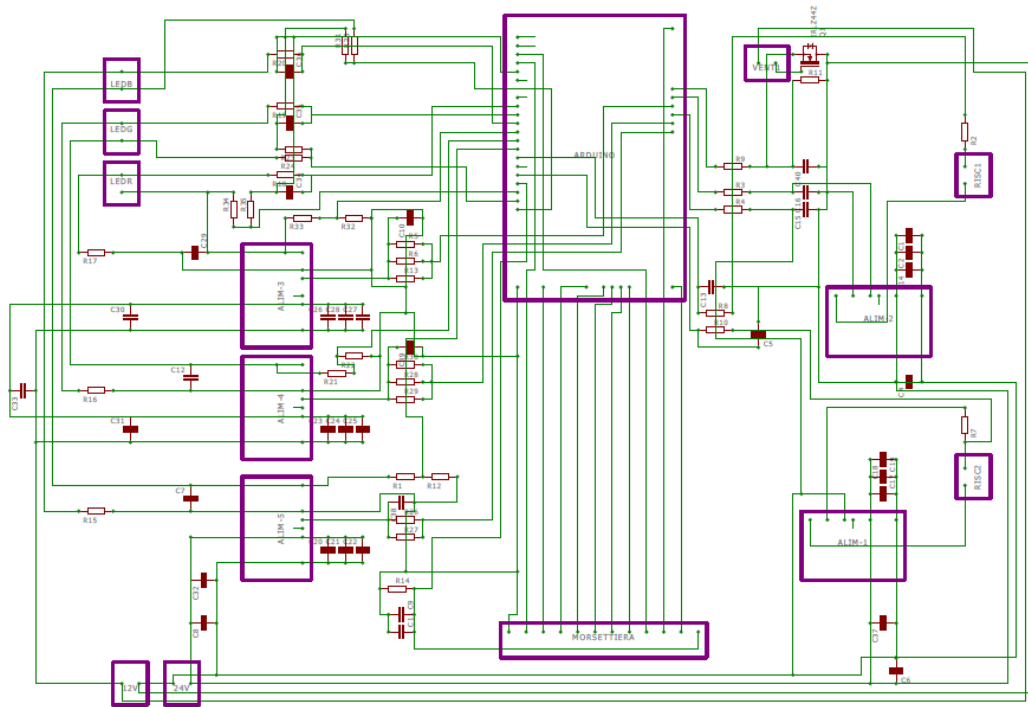
**LeggiSeriale.m:** consente la sola lettura del dato presente sul buffer di Arduino

**Misura\_Tj.m:** consente di misurare la temperatura di giunzione dei LED, utilizzando il metodo descritto nel Capitolo 4, Par. 4.7 – Stima della temperatura di giunzione con il nuovo sistema elettrico.

### *FIRMWARE DI ARDUINO*

Il firmware sviluppato, implementato ed installato all'interno della memoria presente sul circuito Arduino, è in grado di far eseguire le varie tipologie di istruzioni richieste dall'utente attraverso il terminale di controllo TeraTerm. Il file di Arduino caricato all'interno della memoria, al giorno della chiusura della tesi di laurea risulta essere: **LED\_RGB\_23\_02\_2016**

*NUOVO CIRCUITO ELETTRICO SVILUPPATO*



## BIBLIOGRAFIA

- [1] <https://it.wikipedia.org/wiki/Occhio>
- [2] Liem K.F., Bemis W.E., Walker F.V., Grande L. “ *Anatomia comparata degli invertebrati* “ – II Ed., pp. 448 -453 Ed. EDISES – Napoli, 2012
- [3] [http://www.vialattea.net/esperti/php/risposta\\_printable.php?num=11694](http://www.vialattea.net/esperti/php/risposta_printable.php?num=11694)
- [4] Palladino P. “ *Lezioni di illuminotecnica*” – Pp. 25 Ed. Tecniche Nuove – Milano, 2002
- [5] [http://tesi.cab.unipd.it/28287/1/MARCO\\_DAL\\_MONICO\\_tesi.pdf](http://tesi.cab.unipd.it/28287/1/MARCO_DAL_MONICO_tesi.pdf)
- [6] [http://inx.liceicarbonia.it/j7250/images/enrico\\_sailis/Grandezze%20fotometriche.pdf](http://inx.liceicarbonia.it/j7250/images/enrico_sailis/Grandezze%20fotometriche.pdf)
- [7] Appunti del corso di Laurea in ingegneria Elettrica – corso di Illuminotecnica e Fotometria. - Prof. Pietro Fiorentin - Università degli Studi di Padova - 2014
- [8] <http://www.directindustry.it/prod/konica-minolta-sensing/product-13771-25783.html>  
[https://www.pce-instruments.com/italiano/strumento-di-misura/misuratore/luxmetro-pce-instruments-luxmetro-pce-l-100-det\\_138863.htm](https://www.pce-instruments.com/italiano/strumento-di-misura/misuratore/luxmetro-pce-instruments-luxmetro-pce-l-100-det_138863.htm)
- [9] <http://www.serviceinfoweb.com/blog/index.php/il-diodo-led/>
- [10] <http://www.homeled.it/led/>
- [11] <http://ecbgroup.blogspot.it/2015/01/led-per-illuminare.html>
- [12] MohanNed, Undeland Tore. Robbins William P. “*Elettronica di Potenza*” Hoepli Editore – 2005
- [13] M.E. Valcher, M. Bisiacco, “*Controlli Automatici ... tutto quello che avreste voluto sapere a riguardo ma non avete mai osato chiedere*”, Editore - Libreria Progetto.
- [14] Tesi di laurea Magistrale in Ing. Elettrica – Vettoreto A. – “*Realizzazione del controllo di temperatura di una sorgente a cromaticità variabile*” – 2014
- [15] Arduino.cc – Funzionamento e principi fondamentali del Software e Hardware
- [16] F. Bisegna, F. Gugliermetti, M. Barbalace, L. Monti – “*Stato dell’arte dei LED (light Emitting Diodes)*”. Agenzia Nazionale per le Nuove Tecnologie - Giugno 2010
- [18] Shailesh K. R., Ciji Pearl Kurian, Savitha G. Kini – “*Junction temperature measurement of a LED street light using forward voltage method*” – Department of Electrical & Electronics Engineering MIT, Manipal University, Udupi, India . 978-1-4799-5496-4/14/\$31.00 IEEE 2014

[19] [it.mathworks.com/help/matlab](http://it.mathworks.com/help/matlab) – Informazioni fondamentali sul funzionamento e la stesura del codice MatLab